SIEMENS

SINUMERIK

SINUMERIK 828D
Drehen und Fräsen

Inbetriebnahmehandbuch

Gültig für:

CNC-Software Version 4.5 SP2

03/2013

6FC5397-3DP40-3AA1

Rechtliche Hinweise

Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

⚠ GEFAHR
bedeutet, dass Tod oder schwere Körperverletzung eintreten **wird**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ WARNUNG
bedeutet, dass Tod oder schwere Körperverletzung eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

⚠ VORSICHT
bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

ACHTUNG
bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

⚠ WARNUNG
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in dieser Druckschrift werden regelmäßig überprüft, notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten.

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
DEUTSCHLAND

Dokumentbestellnummer: 6FC5397-3DP40-3AA1
Ⓟ 01/2013 Änderungen vorbehalten

Vorwort

SINUMERIK-Dokumentation

Die SINUMERIK-Dokumentation ist in folgende Kategorien gegliedert:

- Allgemeine Dokumentation
- Anwender-Dokumentation
- Hersteller/Service-Dokumentation

Weiterführende Informationen

Unter dem Link (www.siemens.com/motioncontrol/docu) finden Sie Informationen zu folgenden Themen:

- Dokumentation bestellen / Druckschriftenübersicht
- Weiterführende Links für den Download von Dokumenten
- Dokumentation online nutzen (Handbücher/Informationen finden und durchsuchen)

Bei Fragen zur Technischen Dokumentation (z. B. Anregungen, Korrekturen) senden Sie bitte eine E-Mail an folgende Adresse: (mailto:docu.motioncontrol@siemens.com)

My Documentation Manager (MDM)

Unter folgendem Link finden Sie Informationen, um auf Basis der Siemens Inhalte eine OEM-spezifische Maschinen-Dokumentation individuell zusammenstellen: MDM (www.siemens.com/mdm)

Training

Informationen zum Trainingsangebot finden Sie unter:

- SITRAIN (www.siemens.com/sitrain) - das Training von Siemens für Produkte, Systeme und Lösungen der Automatisierungstechnik
- SinuTrain (www.siemens.com/sinutrain) - Trainingssoftware für SINUMERIK

FAQs

Frequently Asked Questions finden Sie in den Service&Support Seiten unter Produkt Support (www.siemens.com/automation/service&support).

SINUMERIK

Informationen zu SINUMERIK finden Sie unter folgendem Link:
(www.siemens.com/sinumerik)

Zielgruppe

Die vorliegende Druckschrift wendet sich an Inbetriebsetzer.

Die Anlage oder das System ist einschaltfertig montiert und angeschlossen. Für die nachfolgenden Schritte, z. B. Konfiguration und Projektierung der einzelnen Komponenten, enthält das Inbetriebnahmehandbuch alle nötigen Informationen oder zumindest Hinweise.

Nutzen

Das Inbetriebnahmehandbuch befähigt die angesprochene Zielgruppe das System oder die Anlage fachgerecht und gefahrlos zu prüfen und in Betrieb zu nehmen.

Nutzungsphase: Aufbau- und Inbetriebnahmephase

Standardumfang

In der vorliegenden Dokumentation ist die Funktionalität des Standardumfangs beschrieben. Ergänzungen oder Änderungen, die durch den Maschinenhersteller vorgenommen werden, werden vom Maschinenhersteller dokumentiert.

Es können in der Steuerung weitere, in dieser Dokumentation nicht erläuterte Funktionen ablauffähig sein. Es besteht jedoch kein Anspruch auf diese Funktionen bei der Neulieferung oder im Servicefall.

Ebenso enthält diese Dokumentation aus Gründen der Übersichtlichkeit nicht sämtliche Detailinformationen zu allen Typen des Produkts und kann auch nicht jeden denkbaren Fall der Aufstellung, des Betriebes und der Instandhaltung berücksichtigen.

Technical Support

Landesspezifische Telefonnummern für technische Beratung finden Sie im Internet unter "Kontakt" (www.siemens.com/automation/service&support).

EG-Konformitätserklärung

Die EG-Konformitätserklärung zur EMV-Richtlinie finden Sie im Internet (www.siemens.com/automation/service&support).

Geben Sie dort als Suchbegriff die Nummer 15257461 ein oder nehmen Sie Kontakt mit der zuständigen Siemens Geschäftsstelle in Ihrer Region auf.

Inhaltsverzeichnis

Lieferumfang und Voraussetzungen 1

1.1 Systemübersicht

Konfiguration mit SINAMICS S120 Booksize

Die folgende Konfiguration zeigt ein typisches Beispiel mit SINAMICS S120 Booksize:

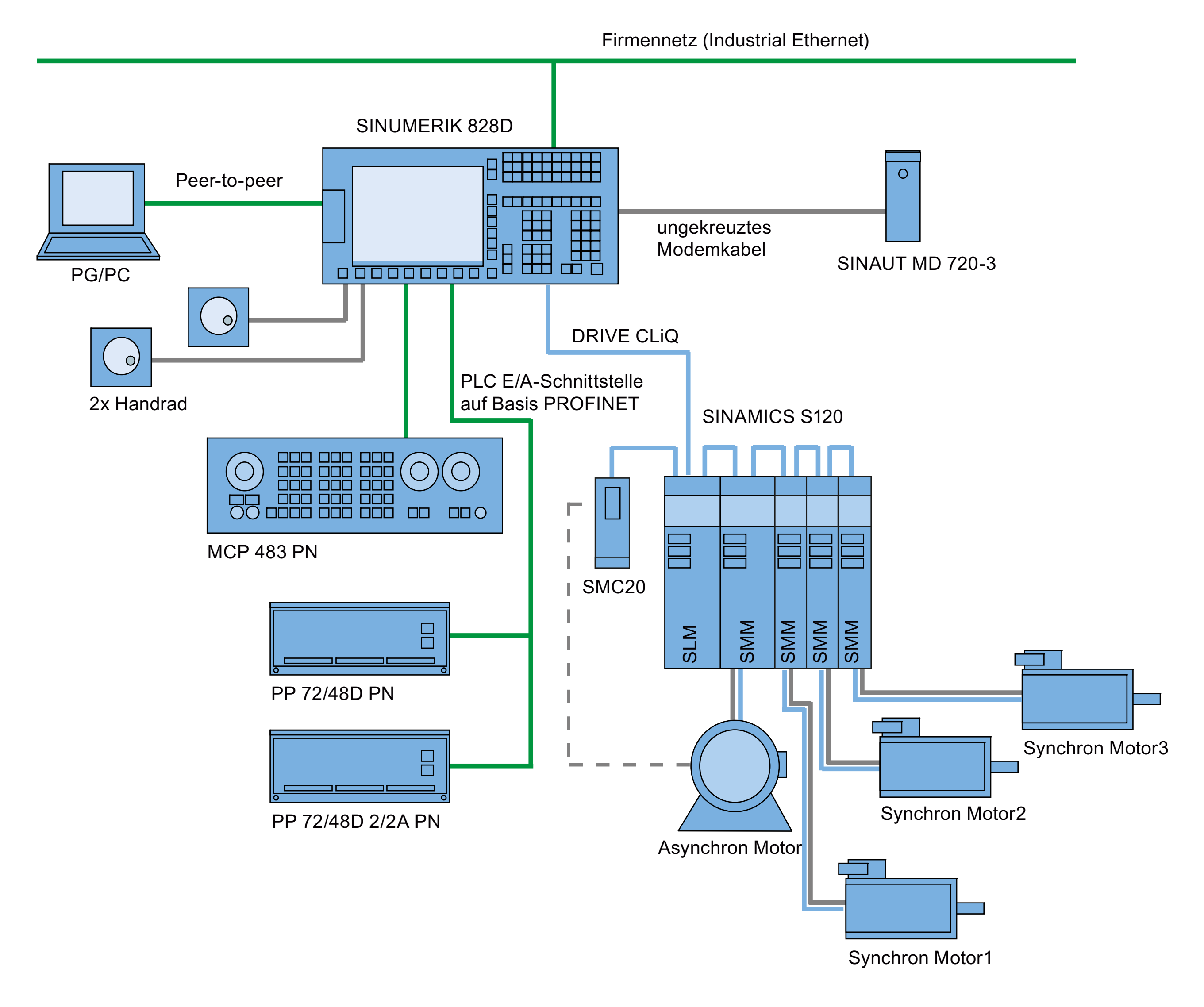


Bild 1-1 Konfigurationsbeispiel 1: Grundausbau mit 4 Achsen

Konfiguration mit SINAMICS S120 Combi

Die folgende Konfiguration zeigt ein Beispiel mit SINAMICS S120 Combi:

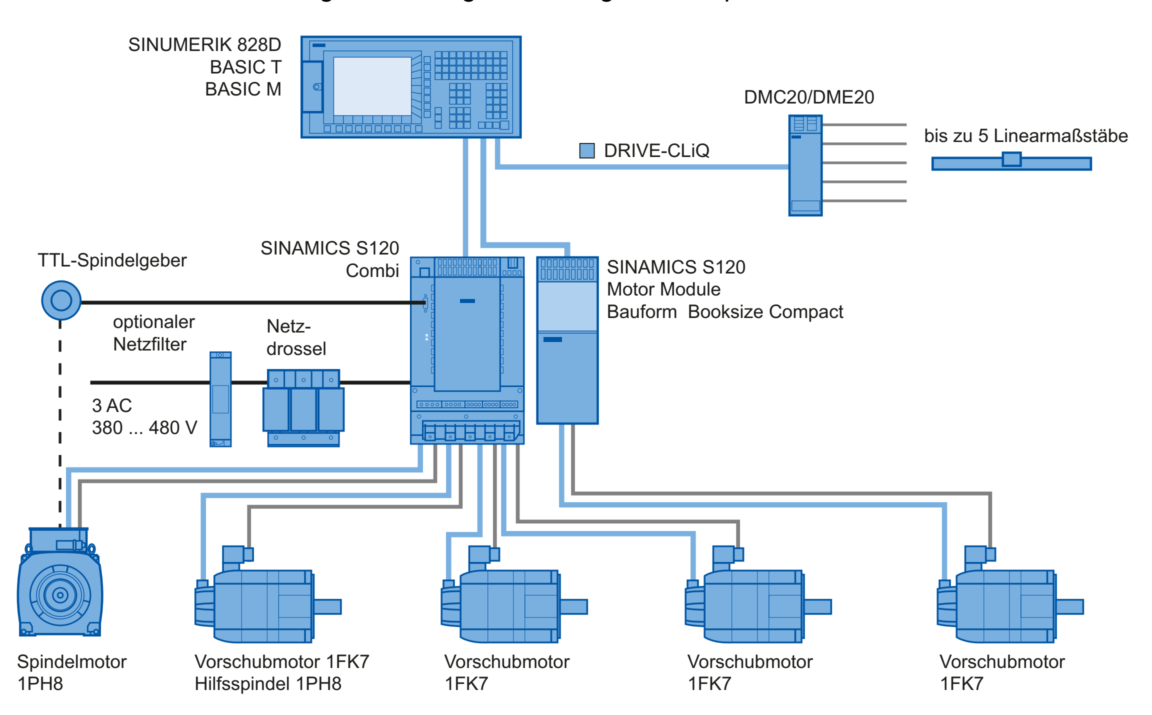


Bild 1-2 Konfigurationsbeispiel 2: SINUMERIK 828D BASIC T/M mit S120 Combi

1.2 Inbtriebnahme- und Service-Tools

Toolbox CD

Die Toolbox CD für SINUMERIK 828D hat folgenden Inhalt:

- PLC Programming Tool for Integrated PLC
- Inbetriebnahme-Software für SINAMICS S120
- PLC Library (Beispiel)

PLC Programming Tool for Integrated PLC

Für die Programmierung der PLC steht folgendes Tool zur Verfügung: PLC Programming Tool for Integrated PLC. Im weiteren Verlauf dieses Handbuchs wird die Bezeichnung "Programming Tool" verwendet.

Inbetriebnahme-Software für SINAMICS S120

Bis zur vollständigen Verfügbarkeit der SINAMICS S120 Inbetriebnahme-Funktionalität über die Bedienoberfläche erfolgt die Antriebskonfiguration und Optimierung über die Inbetriebnahme-Software für SINAMICS S120. Die Verbindung des PC erfolgt über die frontseitige Ethernet-Schnittstelle der SINUMERIK 828D.

Hinweis

Bestelldaten

Die Bestelldaten für die folgenden Tools entnehmen Sie bitte dem Katalog NC 82 und NC 62.

SINUMERIK Integrate Access MyMachine /P2P

Access MyMachine /P2P (früher: RCS Commander) ist ein Tool, mit dem der Inbetriebsetzer sehr einfach per Drag&Drop Dateien zwischen einem PC und der Steuerung austauschen kann.

Zur Datenübertragung wird der PC direkt mit der frontseitigen Ethernet-Schnittstelle verbunden. Für eine Punkt-zu-Punkt Verbindung ist keine zeitintensive Parametrierung der Ethernet-Schnittstelle erforderlich. Alle Einstellungen werden von Access MyMachine /P2P automatisch vorgenommen. Über ein Firmennetzwerk kann mit Access MyMachine /P2P auch sequentiell auf mehrere NCU zugegriffen werden.

Im weiteren Verlauf dieses Handbuchs wird die Bezeichnung "Access MyMachine" verwendet.

Antriebs-/Inbetriebnahme-Software STARTER

Die Antriebsinbetriebnahme der SINUMERIK 828D kann mit dem Antriebs-/Inbetriebnahme-Software STARTER durchgeführt werden. Einfache Inbetriebnahmevorgänge, die üblicherweise vom Feldservice durchgeführt werden (z. B. Aktivierung von direkten Mess-Systemen), können direkt über die Bedienoberfläche der SINUMERIK 828D vorgenommen werden. Erweiterte Inbetriebnahmevorgänge, die üblicherweise bei der Fertigung der Maschine vorgenommen werden (z. B. Optimierung von Antrieben), kann offline über die Inbetriebnahme-Software für SINAMICS S120 durchgeführt werden.

1.3 Ablauf der Inbetriebnahme

1.3.1 Wegweiser durch die Dokumentation

Der in der nachfolgenden Tabelle beschriebene Inbetriebnahmeablauf stellt einen effizienten Vorschlag dar, der auch im SINUMERIK-Training vermittelt wird. Die Angaben in eckigen Klammern beziehen sich auf die Nummern der in den Trainingsunterlagen beschriebenen Schritte. Die in diesem Handbuch enthaltenen Abschnitte, die weiterführende Informationen zu den Trainingsunterlagen enthalten, sind in der rechten Spalte als Verweise angegeben.

Übersicht:

<B060> Systemübersicht	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 11)
<B002> Inbetriebnahme- und Service Tools	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 13)

Grundlagen:

<B061> Menüstruktur	Programmierhandbuch Messzyklen
<B021> Zugriffsstufen	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 43)
<B021> Datum und Uhrzeit	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 46)
<B041> Maschinendaten parametrieren	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 87)
<B032> SINUMERIK Integrate Access MyMachine	Online-Hilfe
<B044> Lizenzen prüfen und eingeben	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 52)
<B045> Netzwerk Konfiguration → Laufwerke einrichten	Bedienhandbuch Drehen Bedienhandbuch Fräsen
<B007> Elektronisches Logbuch	Inbetriebnahmehandbuch "SINUMERIK Operate"
<B017> Datenverwaltung	Inbetriebnahmehandbuch "Drehen und Fräsen" • Einführung und Anwendung von Datenklassen (Seite 33) • Easy Archive (Seite 389)
<B006> Aufbau eines Elektroschranks	Projektierungshandbuch "EMV-Aufbaurichtlinie"

Aufbau und Diagnose:

Thema	Dokumentation
<B010> PPU-Verbindungen Diagnose • Systemübersicht • Digitale Eingänge/Ausgänge	Gerätehandbuch "PPU"
<B030> Digitale Eingänge/Ausgänge → Klemmenbelegung	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 173)
Handräder und Messtaster	
<B051> Line Modules, Motor Modules	Gerätehandbuch SINAMICS S120
<B053> Diagnose • Diagnose Antriebssystem • Diagnose und Service → Achsen	Inbetriebnahmehandbuch "SINUMERIK Operate"
<B008> PROFINET Verbindungen Diagnose: • MCP 310C PN / MCP 483C PN • PP 72/48D PN/PP 72 / 48D 2/2A PN	Gerätehandbuch "PPU"

PLC:

Thema	Dokumentation
<B026> PLC Programm erstellen	Online-Hilfe "PLC Programming Tool"
<B019> S7-200 PLC-Anweisungen	Online-Hilfe "PLC Programming Tool"
<B033> PLC-Funktionen	Funktionshandbuch "Grundlagen" (P4)
<B031> PLC Alarme und Meldungen → Funktionsschnittstelle: PLC-Anwenderalarme	Funktionshandbuch "Grundlagen" (P4)
<B058> PLC Onboard Diagnose	Bedienhandbuch Drehen Bedienhandbuch Fräsen Funktionshandbuch "Grundlagen" (P4)
<B000> PLC Nahtstelle Struktur	Listenhandbuch "NC-Variablen und Nahtstellensignale", Kapitel: "Nahtstellensignale - Übersicht"

Ablauf der Inbetriebnahme:

Thema	Dokumentation
<B068> Inbetriebnahme Ablauf → Antrieb in Betrieb nehmen	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 103)
<B084> Referenzieren Geberjustage	Funktionshandbuch "Grundlagen" (R1)
<B101> Antriebsoptimierung	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 181)
<B102> Werkzeugverwaltung	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 305)
<B009> Easy Extend	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 241)

<B011>	SINUMERIK Integrate Run MyScreens	Programmierhandbuch "SINUMERIK Integrate Run MyScreens" (BE2)
<B020>	Service Planner	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 227)
<B018>	Easy Message	Bedienhandbuch Drehen Bedienhandbuch Fräsen
<B016>	Installation einer zusätzlichen Achse	---
<B029>	Analog-Spindel	Inbetriebnahmehandbuch "Drehen und Fräsen" (Seite 96)

Randbedingungen

In diesem Handbuch wird von folgenden Randbedingungen ausgegangen:

- Die mechanische und elektrische Montage der Anlage ist abgeschlossen.
- Sichtprüfung der Anlage auf:
  - Korrekten mechanischen Aufbau mit festen elektrischen Verbindungen
  - Anschluss der Spannungsversorgung
  - Anschluss der Schirmung und Erdung
- Einschalten der Steuerung und Hochlauf im "Normal startup":

  Der Hochlauf der Steuerung ist beendet, wenn das Grundbild der Bediensoftware angezeigt wird.

Literatur

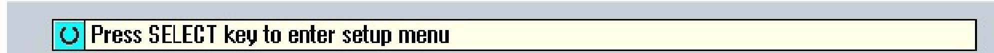
Einzelheiten zur Inbetriebnahme der Zyklen unter SINUMERIK Operate finden Sie im Inbetriebnahmehandbuch Bediensoftware und Basesoftware; SINUMERIK Operate (IM9), Kapitel "Zyklen konfigurieren".

1.4 Hochlauf der Steuerung

Steuerungshochlauf

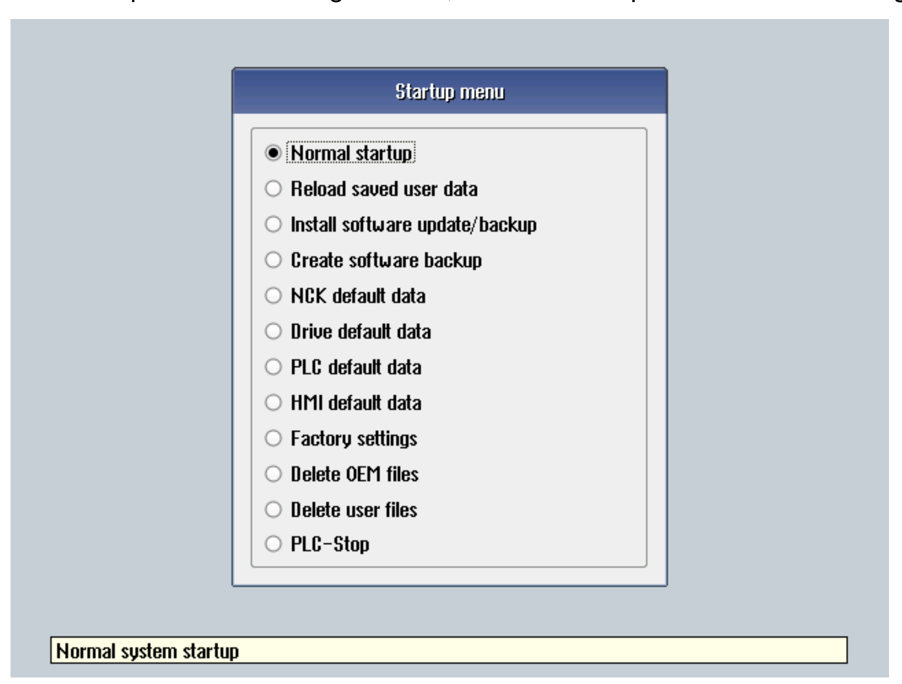
Vorgehensweise:

1. Schalten Sie die Steuerung ein, dann erscheint beim Hochlauf folgende Anzeige:



2. Betätigen Sie innerhalb von 3 Sekunden die Taste <SELECT>.
3. Drücken Sie danach folgende Tasten nacheinander:

   Menürückschalt-Taste, **HSK2** (horizontaler SK2), **VSK2** (vertikaler SK2)
4. Das "Setup menu" wird aufgeblendet, "Normal startup" ist die Voreinstellung.



Betriebsarten für den Hochlauf

Auswahl	Funktion
Normal startup	Das System führt einen normalen Hochlauf durch.
Reload saved user data	Das System lädt die gespeicherten Anwenderdaten (Softkey "Daten sichern") von der System CompactFlash Card.
Install software update/backup	Es wird ein Update auf die System CompactFlash Card von der Anwender CompactFlash Card oder USB-FlashDrive installiert.
Create software backup	Es wird ein Backup von der System CompactFlash Card auf Anwender CompactFlash Card oder USB-FlashDrive gespeichert.

Auswahl	Funktion
NCK default data	Das System lädt die bei Auslieferung eingestellten Siemens NCK-Daten und löscht die remanenten Daten auf der PLC.
Drive default data	Die SINAMICS Anwenderdaten werden gelöscht.
PLC default data	PLC urlöschen und Default NOP PLC-Programm laden.
HMI default data	Die Anwenderdaten im Verzeichnis /user werden gelöscht.
Factory settings	Auswahl zwischen zwei Fällen: **No** [Fall 1]/ **Yes** [Fall 2] • **Fall 1:** Die SINAMICS Anwenderdaten werden gelöscht. Siemens Standarddaten NCK werden geladen. PLC urlöschen und Default NOP PLC-Programm laden. Die Anwenderdaten im Verzeichnis /user werden gespeichert. • **Fall 2:** Wie Fall 1 und zusätzlich: Löschen der Daten in den Verzeichnissen /oem und /addon.
Delete OEM files	Es werden alle Daten unter /oem und /addon gelöscht: Archive Hersteller; OEM Alarmtexte; Easy Screen Applikation.
Delete user files	
PLC stop	PLC wird in Stopp gesetzt.

Hinweis

Tausch der System CompactFlash Card zwischen verschiedenen PPUs

Durch die bei SINUMERIK 828D systembedingte Abhängigkeit zwischen CompactFlash Card und SRAM bei der Datenhaltung, ist die System CompactFlash Card wie ein fest eingelöteter EEPROM zu betrachten und sollte nicht getauscht werden!

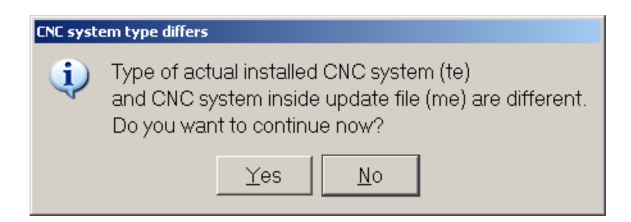
Tritt dieser Fall aus zwingenden Gründen auf, wird im Hochlauf der Tausch der System CompactFlash Card anhand der hinterlegten Seriennummer erkannt.

Die Reaktion der Steuerung ist ein Laden von gesicherten Daten im Hochlauf (Sicherung erfolgte zuvor manuell mit Softkey "Save Data"). Werden keine gesicherten Daten gefunden, so wird automatisch ein Hochlauf mit "NCK default data" ausgeführt.

Überprüfung der Hardware

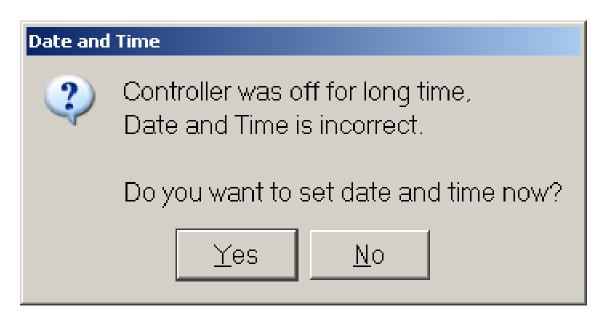
Bei der Hardware PPU 24x.2 wird zwischen der Drehvariante BASIC T und der Fräsvariante BASIC M unterschieden:

Diese Unterscheidung erfolgt im Hochlauf anhand der im Hardware-Infoblock vorgefundenen MLFB und wird mit der vorhandenen NCK-Skalierung verglichen. Wird eine nicht für diese Hardware geeignete Software gefunden, wird folgende Meldung ausgegeben:

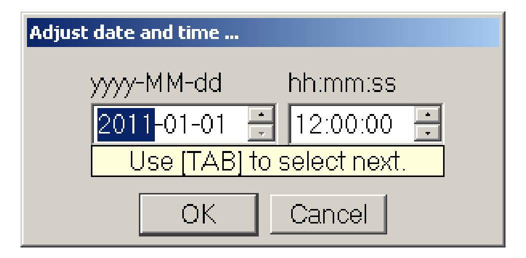


Leerer RTC-Kondensator

Wenn der RTC-Kondensator entladen ist, wird im Hochlauf folgende Meldung ausgegeben:



Danach haben Sie die Möglichkeit, Datum und Uhrzeit neu einzustellen:



Nachdem Hochlauf während die Steuerung eingeschaltet ist, wird der Kondensator wieder aufgeladen.

1.5 Kommunikation mit der Steuerung

Aufbau der Verbindung

Für den Verbindungsaufbau zwischen Steuerung und PG/PC wird ein Ethernet-Kabel benötigt. An der Steuerung stehen folgende Ethernet-Schnittstellen zur Verfügung:

- Anschluss über X127 (hinter der Klappe auf der Frontseite):

  Kabeltyp: gekreuztes Ethernet-Kabel

  An der Schnittstelle X127 ist die Steuerung als DHCP-Server voreingestellt, der für eine Direktverbindung (Peer-to-peer Anschluss) die IP-Adresse 192.168.215.1 liefert.

- Anschluss über X130 (Rückseite):

  Kabeltyp: nicht gekreuztes Ethernet-Kabel

  Die Schnittstelle X130 ist der Anschluss an das Firmennetz. Die IP-Adresse, die das PG/PC als DHCP-Client hier erhält, bestimmt der DHCP-Server aus dem Firmennetz oder es wird manuell eine feste IP-Adresse eingetragen.

1.5.1 Kommunikation mit der Steuerung über X130

Anschluss an das Firmennetz

Die NCU ist über die Ethernet-Schnittstelle X130 an das Firmennetz angeschlossen. Über das Firmennetz wird z. B. auf die Netzlaufwerke zugegriffen.

Wählen Sie im Bedienbereich "Diagnose" mit der Menüfortschalt-Taste den Softkey "Bus TCP/IP " → "TCP/IP Diagnose" → "Details" an, um die Parameter für die Kommunikation über X130 einzustellen.

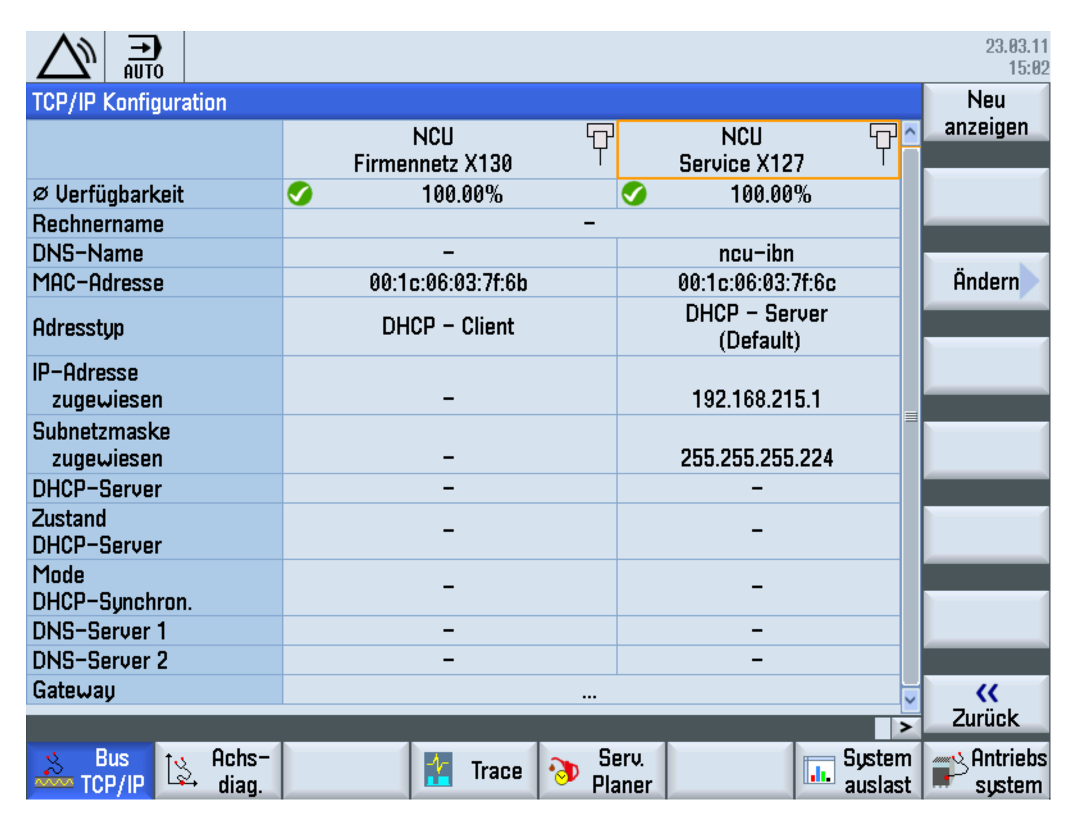


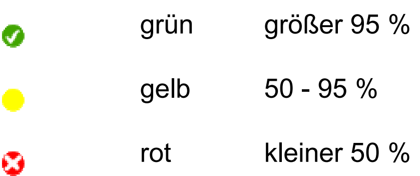
Bild 1-3 Netzwerk-Einstellungen

Firmennetz X130

	weiß	Netzwerkkabel gesteckt
	rot	Netzwerkkabel nicht gesteckt

Verfügbarkeit

Die Verfügbarkeit beschreibt den prozentualen Anteil an fehlerhaften Daten gemessen am gesamten Datenvolumen. Probleme im Firmennetz (z. B. logische Laufwerke, die nicht erreichbar sind, doppelte IP-Adresse etc.) sowie die Einschwingzeit während des Hochlaufs können zu Schwankungen bei der Verfügbarkeit führen:



	grün	größer 95 %
	gelb	50 - 95 %
	rot	kleiner 50 %

Hinweis

Alle nicht vorhandenen Informationen werden in der entsprechenden Tabellenzelle mit einem Bindestrich "-" gekennzeichnet.

1.5.2 So kommunizieren Sie über das Programming Tool mit der Steuerung

Einstellung der Kommunikationsschnittstelle im PLC Programming Tool

Zum Einrichten der Netzwerkverbindung im PLC Programming Tool gehen Sie folgendermaßen vor:

1. Starten Sie das PLC Programming Tool über folgende Verknüpfung oder über das Menü "Start":

2. Klicken Sie in der Navigationsleiste auf das Symbol "Kommunikation" oder wählen Sie den Menübefehl "Ansicht" → "Kommunikation".
3. Geben Sie in der linken Spalte unter "Kommunikationsparameter" als IP-Adresse 192.168.215.1 ein für X127.
4. Doppelklicken Sie rechts oben auf das Symbol "TCP/IP".
5. Wählen Sie im Dialog "PG/PC-Schnittstelle" das TCP/IP-Protokoll des PG/PC aus. Das ist normalerweise die Netzwerkkarte des PC.

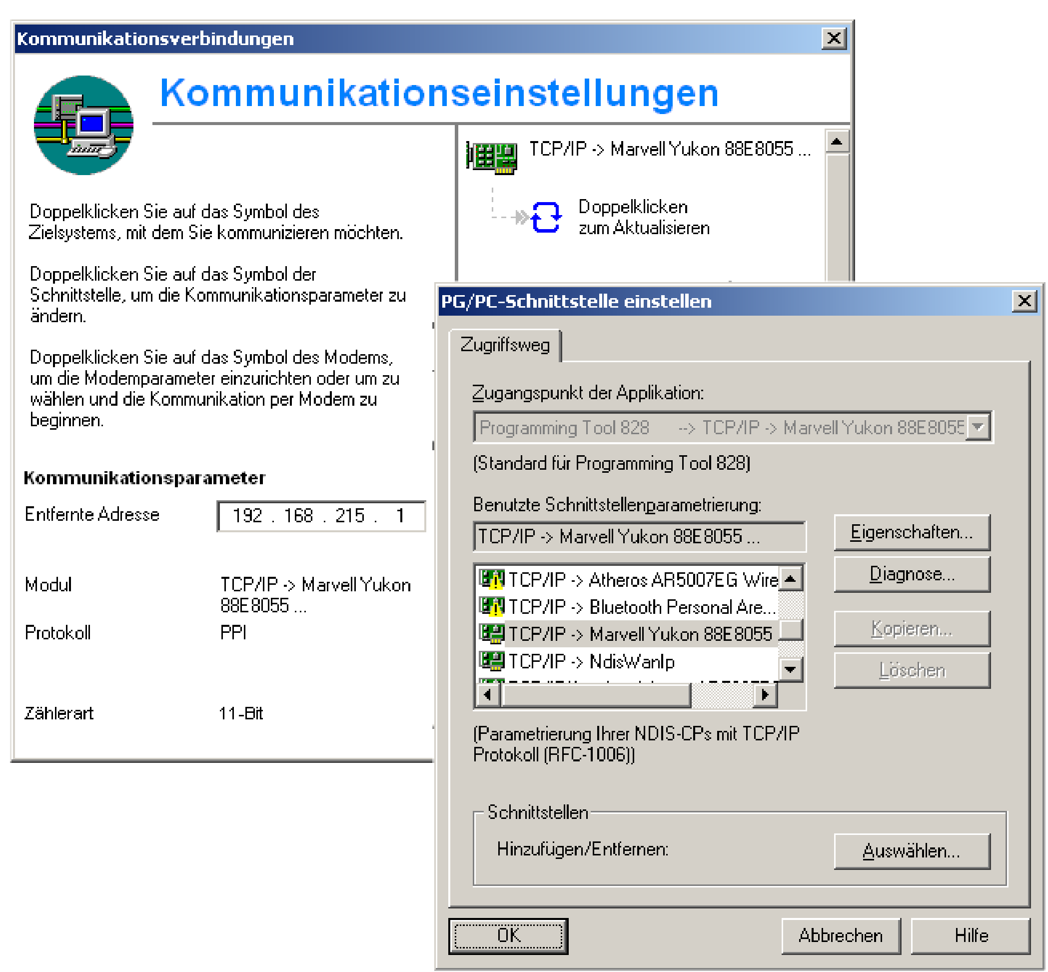


Bild 1-4 TCP/IP Kommunikationseinstellungen

6. Bestätigen Sie mit "OK".
7. Doppelklicken Sie auf das Symbol "Doppelklicken zum Aktualisieren", um die Verbindung herzustellen. Wird die Verbindung erfolgreich aufgebaut, wird das Symbol mit einem grünen Rand dargestellt:

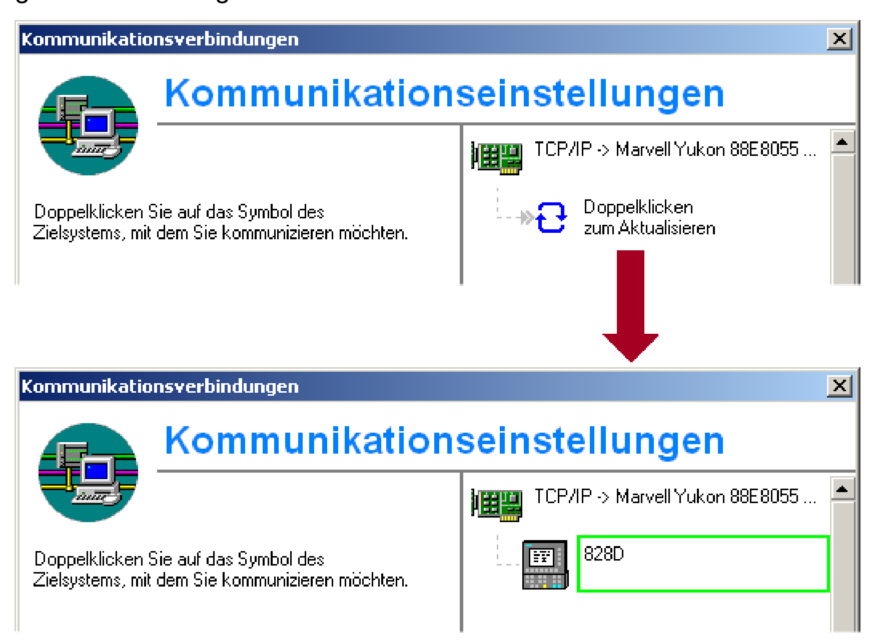


Bild 1-5 Online-Verbindung

8. Falls die Verbindung nicht zustande kommt, muss eventuell noch folgende Einstellung deaktiviert werden:

   Wählen Sie "Systemsteuerung" → "Netzwerkverbindungen" → "Local Area Connection" "Eigenschaften" → "Erweitert" → "Windows Firewall" → "Einstellungen" → "Erweitert ": Deaktivieren Sie die Option "Local Area Connection".

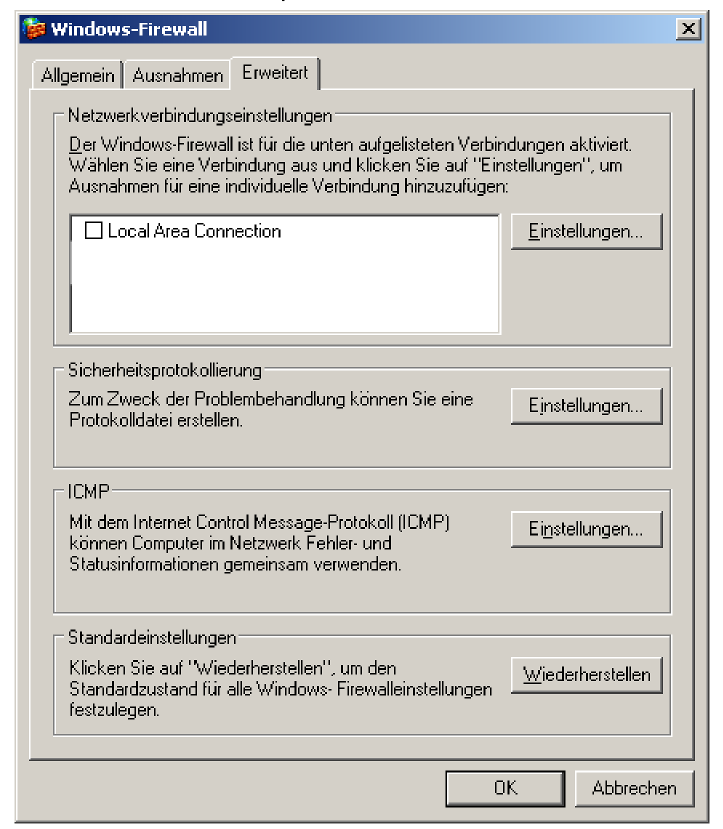


Bild 1-6 Option deaktivieren

Bestätigen Sie mit "OK" und wiederholen Sie Schritt 7.

1.5.3 So kommunizieren Sie mit der Steuerung über Access MyMachine

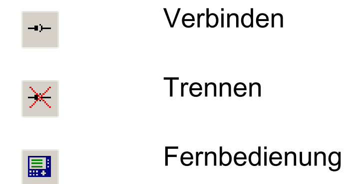
Verbindungsmöglichkeiten

Um eine Verbindung mit der Steuerung herzustellen, stehen für "Access MyMachine" folgende Möglichkeiten zur Verfügung:

- Direktverbindung (Peer-to-peer)
- Netzwerkverbindung

Der aktuelle Zustand der Verbindung wird unten in der Statuszeile von Access MyMachine angezeigt.

Bedeutung der Schaltflächen:



Verbinden

Trennen

Fernbedienung

Hinweis

Es ist generell nur **eine Verbindung** zulässig, d.h. mehrere Verbindungen gleichzeitig zu verschiedenen Steuerungen werden nicht unterstützt: Somit ist ein Datenaustausch zwischen zwei Steuerungen mittels "Access MyMachine" ausgeschlossen.

Direktverbindung

Um eine Direktverbindung herzustellen:

1. Die Login-Daten werden Dialog "Einstellungen" → "Verbindung" → "Direktverbindung" vorgenommen:

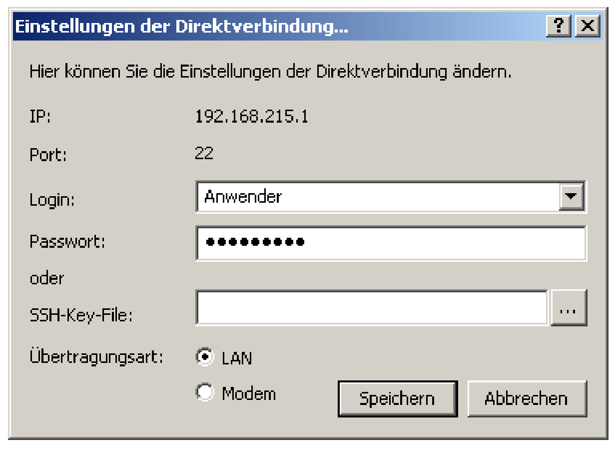


Bild 1-7 Dialog: Login-Daten für Direktverbindung

2. Wählen Sie im Menü "Verbindung" → "Verbinden" → "Direktverbindung" oder klicken Sie auf die Schaltfläche "Verbinden".

   Der folgende Dialog wird angezeigt:

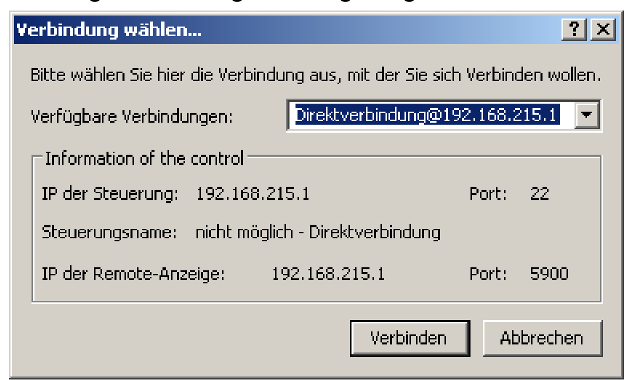


Bild 1-8 Dialog: Direktverbindung

3. Die letzte gewählte Direktverbindung ist markiert. Mit der Schaltfläche "Verbinden" wird die Verbindung zu IP-Adresse 196.168.215.1 hergestellt.

   Dieser Dialog erscheint nicht, wenn man über das Menü die Direktverbindung anwählt.

Netzwerkverbindung

Um eine Netzwerkverbindung herzustellen:

1. Wählen Sie im Menü "Einstellungen" → "Verbindung" → "Direktverbindung" oder klicken Sie auf die Schaltfläche "Verbinden".

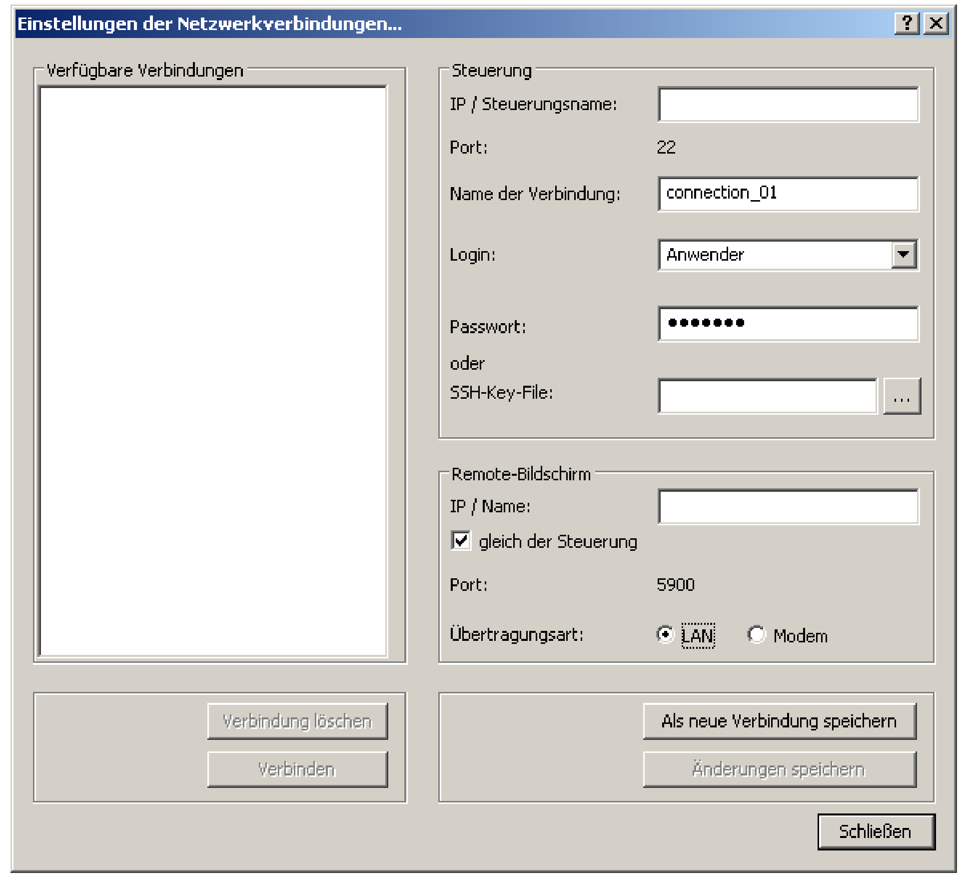


Bild 1-9 Dialog: Netzverbindung

2. Wählen Sie im Menü "Verbindung" → "Verbinden" → "Netzwerkverbindung" oder wählen Sie - falls vorhanden - eine der zuletzt verwendeten Verbindungen an.
3. Verbindungsaufnahme zur parametrierten Steuerung erfolgt.

Hinweis

SSh-Key File

Alternativ zur Eingabe eines Passworts kann der Benutzer zur Authentifikation auch einen SSh-Schlüssel verwenden. Weitere Informationen hierzu finden Sie in der Online-Hilfe.

1.5.4 Beispiel: So kommunizieren Sie über den NCU Connection Wizard mit der Steuerung

Voraussetzungen

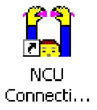
Die Inbetriebnahme-Software für SINAMICS S120 ist auf PG/PC installiert. Der "NCU Connection Wizard" ist Bestandteil dieser Software.

Die Verbindung zur Steuerung ist über das PLC Programming Tool bereits eingerichtet.

Verbindung zur Steuerung herstellen

Vorgehensweise auf dem PG/PC:

1. Starten Sie "NCU Connection Wizard" über diese Verknüpfung oder über das Menü "Start":



2. Im Dialog "Select Control Model" wählen Sie für die Verbindung zur SINUMERIK 828D als NCU Typ "840D solution line" aus.

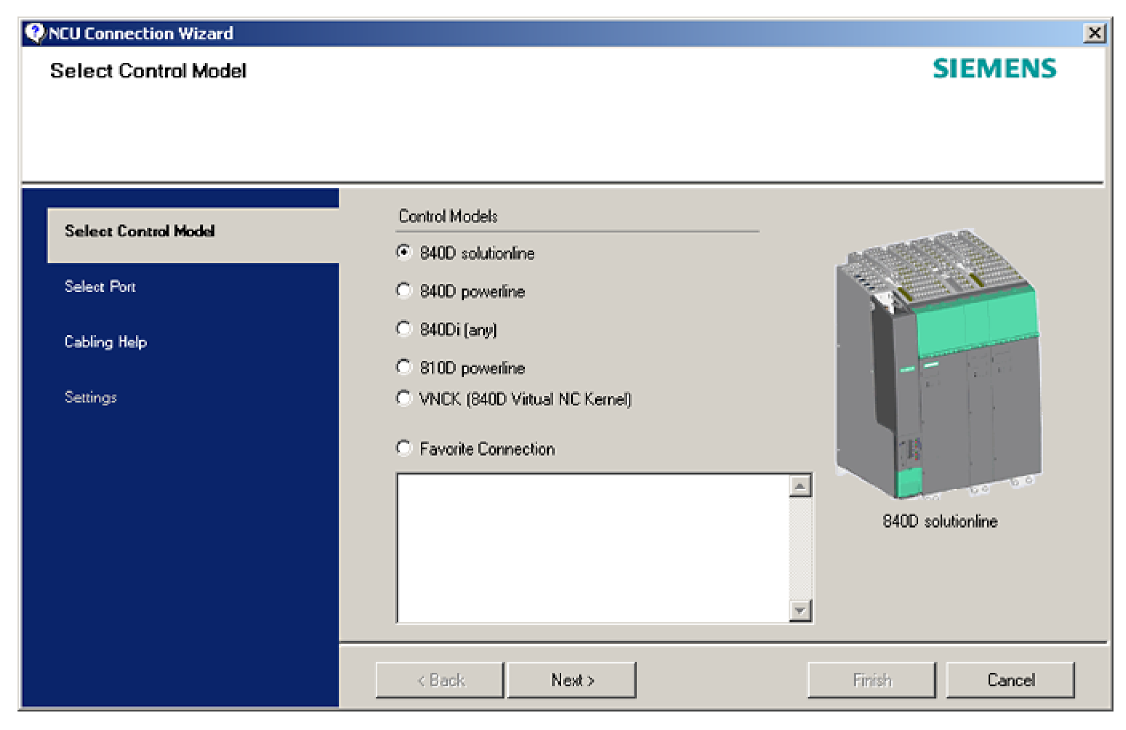


Bild 1-10 NCU Typ wählen

3. Im Dialog "Select Port" wählen Sie den Anschluss an die Steuerung, mit dem Sie sich über Ethernet verbunden haben.

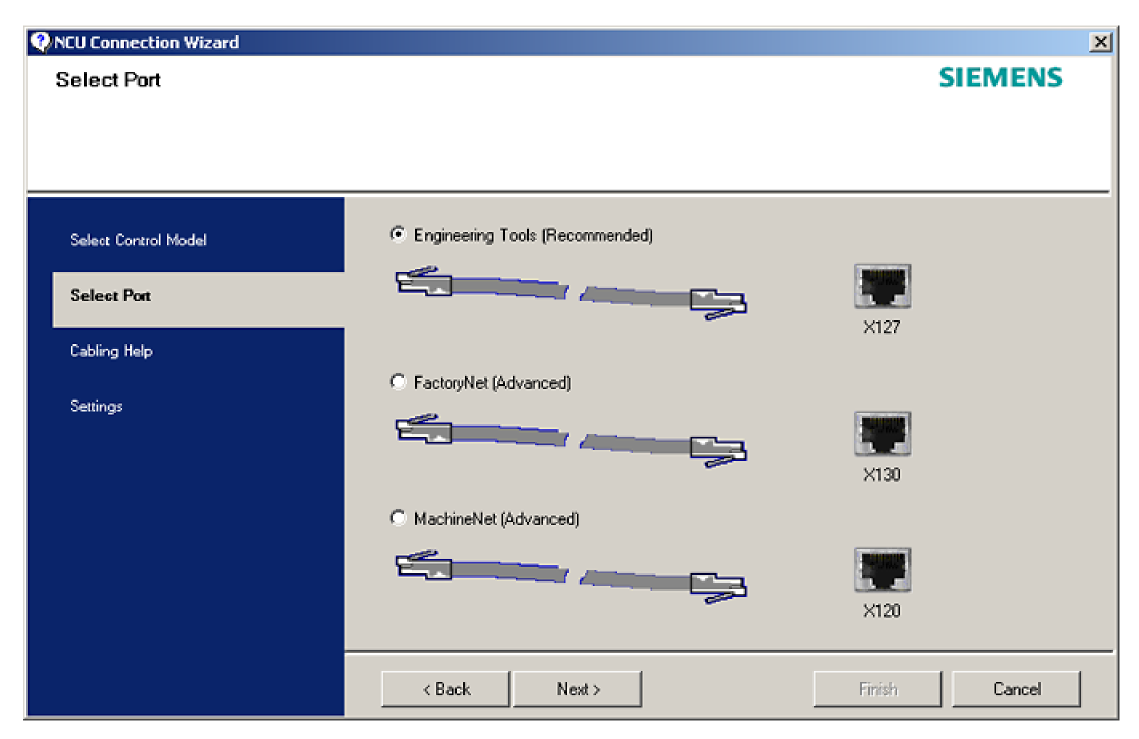


Bild 1-11 Anschluss auswählen

4. Im Dialog "Cabling Help" bestätigen Sie die Kabelverbindung der beiden Geräte.

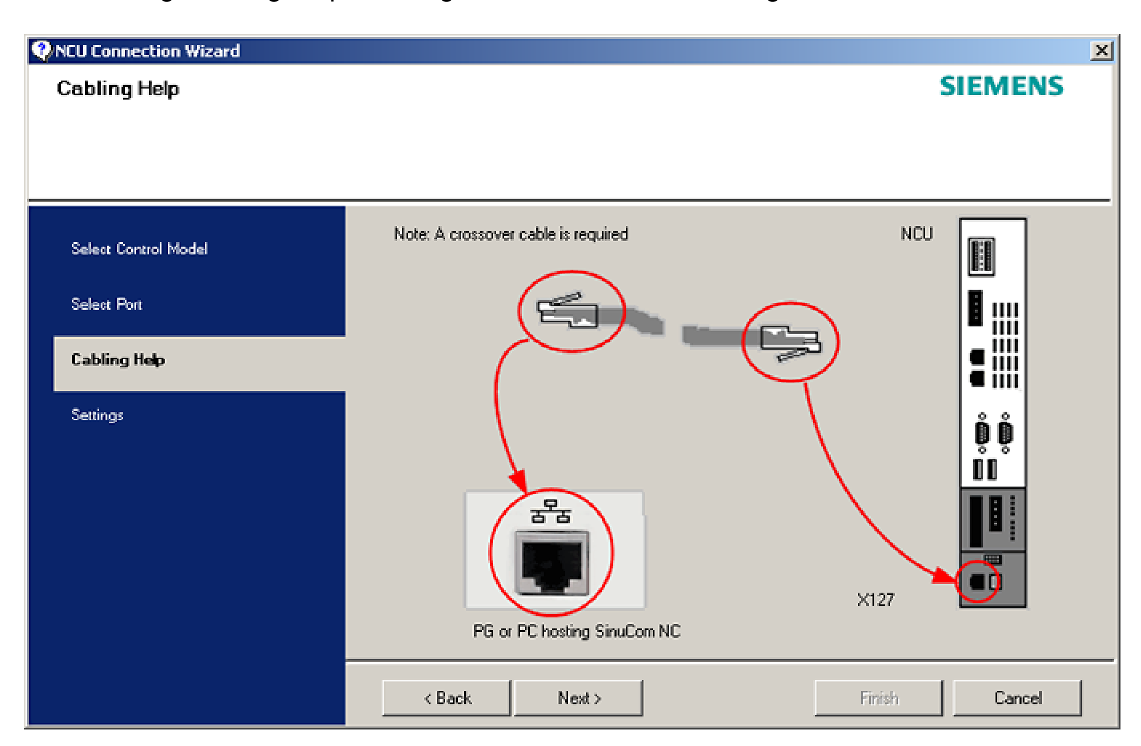


Bild 1-12 Verkabelung

5. Im Dialog "Settings" prüfen Sie die IP-Adresse und geben einen Namen für diese Einstellungen an.

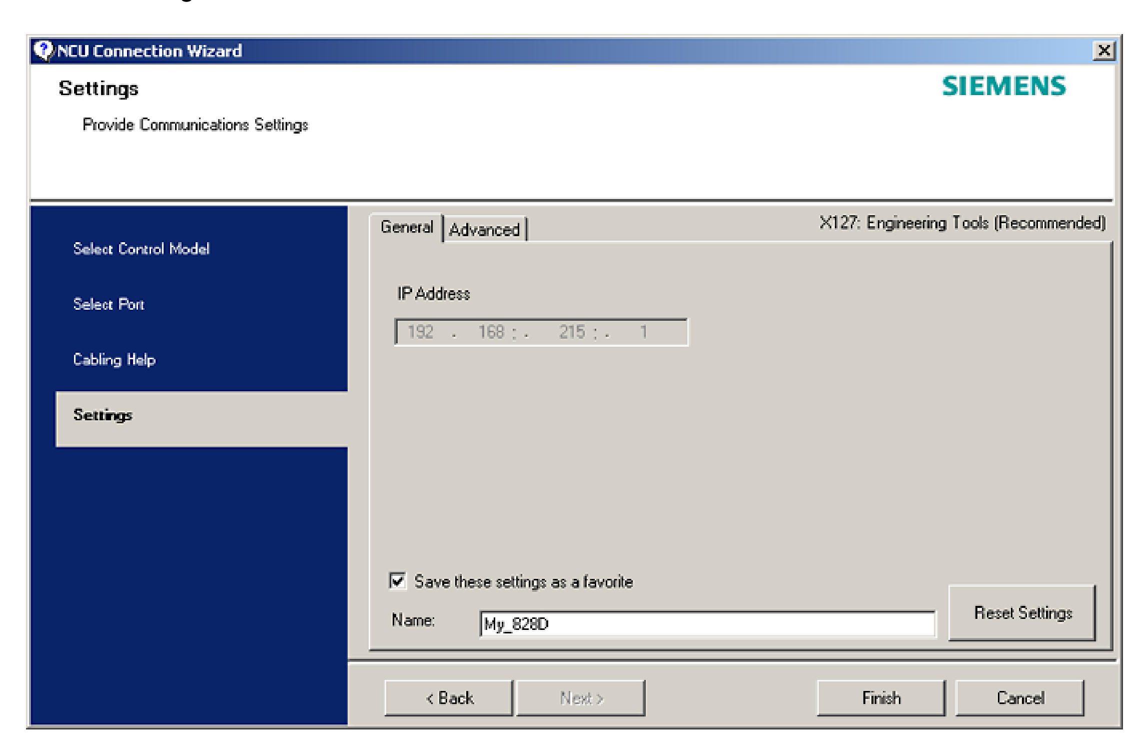


Bild 1-13 Netzwerk Einstellungen

2 Einführung und Anwendung von Datenklassen

Zielsetzung

Das Prinzip der Datenklassen schafft eine klare Trennung zwischen Systemdaten, (Daten, die bei der Installation der Software entstehen), OEM-Daten (Daten, die bei der Inbetriebnahme einer Prototyp-Maschine entstehen) und User-Daten (Daten, die beim Anwender entstehen). Die OEM-Daten werden weiter unterschieden in Daten, die für eine Maschinenserie gelten (einheitliche Daten an allen Maschinen einer Serie) und den individuellen Daten, die nur für eine bestimmte Maschine gelten.

Diese Klassifizierung von Daten findet bei der Erstellung von Datenklassenarchiven, also Archiven die lediglich einen Untermenge der Daten (Herstellerdaten, maschinen-individuelle Daten oder Anwenderdaten) enthalten, Verwendung. Damit werden Inbetriebnahme und Hochrüstung erleichtert.

2.1 Datenklassen im NCK

Einteilung

Alle steuerungsrelevanten Daten werden mit Hinblick auf die spätere Verwendung zur Inbetriebnahme, Software-Upgrade, Software-Update und Komponententausch in Datenklassen eingeteilt. Die Daten sind in vier Datenklassen eingeteilt: M (Manufacturer), I (Individual), U (User) und S (System)

Datenklasse System (S)

Diese Datenklasse umfasst die Daten in den Verzeichnissen Siemens und System auf der CompactFlash Card und wird im weiteren Dokument mit "S" bezeichnet.

Kennzeichen / Eigenschaften der Datenklasse S

Die Daten werden beim ersten Einschalten des Systems oder bei einer Initialisierung automatisch geladen und soweit erforderlich aktiviert.

Die Daten der Klasse "S" können nicht gesichert werden und sind als "read only" gekennzeichnet.

Datentyp	Beispiel
HMI:	Siemens Standard-Bedienoberfläche und deren Spracherweiterungen
NCK:	von Siemens bereitgestellte Standardzyklen, Messzyklen und JobShop-Zyklen, Definitionen wie SGUD und SMAC und alle weiteren NCK aktiven Daten der Datenklasse S
PLC:	Hardware-Konfigurationen
Antriebe:	Code und Daten (Datenbeschreibungen, Alarmbeschreibungen, Makros)

Datenklasse Manufacturer (M)

Diese Datenklasse umfasst alle Daten, die vom Maschinenhersteller (OEM) bei der Erstinbetriebnahme einer Maschine einer Baureihe festgelegt werden, und wird im weiteren Dokument mit "M" bezeichnet.

Kennzeichen / Eigenschaften der Datenklasse M

Daten der Klasse M werden in einem separaten Archiv "Manufacturer" gesichert.

Datentyp	Beispiel
HMI:	OEM-Dialoge (Easy Screen) und Zyklenunterstützung, Alarm- und Meldetexte.
NCK:	Herstellerzyklen, Definitionen MGUD und MMAC und alle NC-aktiven Daten der Datenklasse M (keine Speicher normierenden Maschinendaten)
PLC:	PLC-Programm, Inhalte der Datenbausteine dieser Datenklasse
Antriebe:	alle

Datenklasse Individual (I)

Diese Datenklasse umfasst die Daten, die eine bestimmte Maschine betreffen und im Rahmen der Inbetriebnahme ebenfalls vom OEM oder zu einem späteren Zeitpunkt beim Händler erzeugt werden. Diese Datenklasse wird im weiteren Dokument mit "I" bezeichnet.

Kennzeichen / Eigenschaften der Datenklasse I

Daten der Klasse I werden in einem separaten Archiv INDIVIDUAL gesichert.

Datentyp	Beispiel
HMI:	Easy Extend und Service Planner
NCK:	Referenzpunktmaße, Losekorrektur, Kompensationsdaten, Werkzeugträgerdaten, alle NC-aktiven Daten der Datenklasse I
PLC:	Inhalte der Datenbausteine dieser Datenklasse
Antriebe:	Zurzeit sind keine Daten vorgesehen.

Datenklasse User (U)

Diese Datenklasse umfasst alle Daten des Anwenders sowie die Daten, die zur Laufzeit der Maschine entstehen, beispielsweise der Wartungsintervall-Timer. Diese Datenklasse wird im weiteren Dokument mit "U" bezeichnet.

Kennzeichen / Eigenschaften der Datenklasse U

Daten der Klasse U werden in einem separaten Archiv "User" gesichert.

Datentyp	Beispiel
HMI:	---
NCK:	Werkzeugdaten, Setting-Daten, Teileprogramme, Unterprogramme und Anwender-Zyklen, Definitionen UGUD und UMAC, jedoch kein Programmcode (z. B. PLC)
PLC:	Inhalt der Datenbausteine dieser Datenklasse
Antriebe:	Zurzeit sind keine Daten vorgesehen.

GUD und Makros

Die Zuordnung der Definitionsdateien (GUD, Makros) zu einer Datenklasse erfolgt gemäß folgender Tabelle:

Definitisonsdatei	Datenklasse
GUD4	Individual (I)
MGUD	Manufacturer (M)
MMAC	Manufacturer (M)
UGUD	User (U)
UMAC	User (U)

Datenklassenattribut

Das Datenklassenattribut ist implizit gesetzt. Die Voreinstellung des Datenklassenattributs kann explizit geändert werden.

- Im Allgemeinen werden Daten aus dem aktiven Filesystem aufgrund ihrer Zugehörigkeit zu einer bestimmten Datei **einer Datenklasse** zugeordnet.

  Beispiel: Kompensationsdaten (Daten aus CEC, EEC oder QEC) sind der Datenklasse "Individual" zugeordnet. Werden Daten durch Definition erzeugt (GUD und Makros) so erben diese Daten die Datenklasse in der sich die Definitionsdatei befindet. GUD aus UGUD.DEF werden der Datenklasse "User" zugeordnet.
- Sollen sich jedoch Datendefinition und Dateninhalt **in verschiedenen Datenklassen** befinden, so muss dies in der Datendefinition angegeben werden.

  Beispiel: Die Definition des GUD, das einen Messtaster beschreibt, soll sich in der Datenklasse "Manufacturer" befinden, da es für den Ablauf der Herstellerzyklen notwendig ist.

  Der Wert des Datums soll aber, da der Typ des Messtasters von Maschine zu Maschine unterschiedlich sein kann, der Datenklasse "Individuell" angehören. Dazu wird das zusätzliche Variablenattribut "DC" (DataClass) eingeführt:

  ```
  MGUD.DEF (Datenklasse "Manufacturer")

  DEF CHAN DCI INT CALIPER
  ```

Die Definition der Datenklasse des Dateninhalts kann nur "kleiner oder gleich" der Datenklasse der Definition sein. Anderslautende Projektierungen werden mit einem Alarm abgelehnt.

Dazu wird folgende Wertigkeit der Datenklassen angenommen: **M > I > U**

2.2 Datenklassen in der PLC

Datenklassen in der PLC

Die Zuordnung von Programmbausteinen in Datenklassen dient zielgerichtet einer Hochrüstung oder Inbetriebnahme einer SINUMERIK Steuerung. Hierbei werden drei Datenklassen unterschieden:

- **Manufacturer (M)** wird vom Maschinenhersteller bei der Konstruktion vergeben.
- **Individual (I)** wird vom Maschinenhersteller bei der Erstinbetriebnahme vergeben (Werte für diese spezielle Maschine).
- **User (U)** wird vom Endkunden eingestellt (Werte des jeweiligen Prozesses). Dem Haupt- und Unterprogrammen kann keine andere Datenklasse zugeordnet werden.

Baustein	Voreinstellung	änderbar
Hauptprogramm (OB1)	MANUFACTURER	nein
Unterprogramm (SBR, INT, ...)	MANUFACTURER	nein
Datenbaustein	INDIVIDUAL MANUFACTURER USER	ja

Beispiel

Die Datenklasse wird dem Baustein im Dialog "Eigenschaften" zugewiesen:

1. Wählen Sie im Menü "Ansicht" → "Datenbaustein" einen Datenbaustein aus.
2. Wählen Sie im Menü "Ansicht" → "Eigenschaften", um die Datenklasse zu ändern.

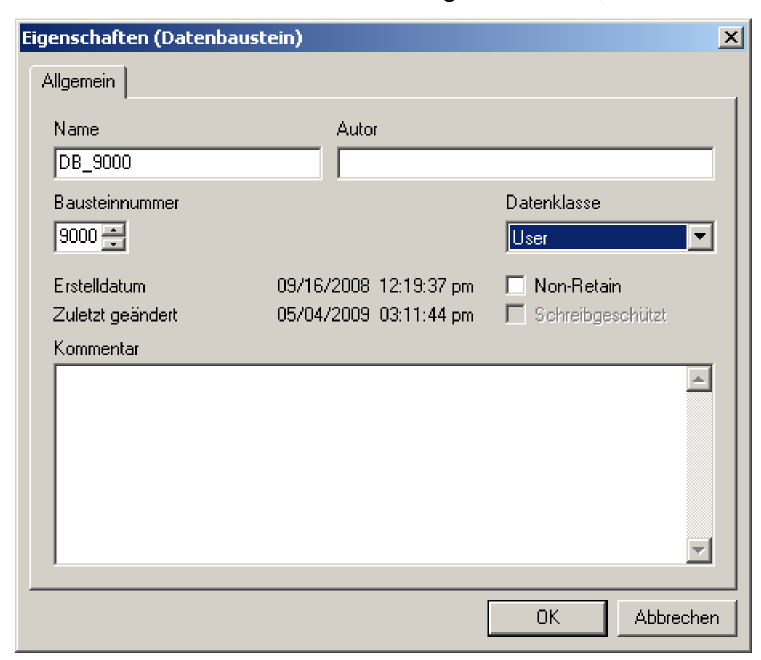


Bild 2-1 Datenklasse wählen

2.3 Datenklassen in der Bediensoftware

Darstellung der Datenklassen in der Bediensoftware

Die Gliederung in Datenklassen wird im Wesentlichen bereits durch die Verzeichnisstruktur der CompactFlash Card wiedergegeben:

Verzeichnis	Datenklasse
system siemens	SYSTEM
addon	MANUFACTURER
oem	MANUFACTURER, INDIVIDUAL
individual	MANUFACTURER, INDIVIDUAL
user	USER

Die Verzeichnisse "System" und "Siemens" haben für die Archivierung keine Bedeutung, da sie mit der Installation der SINUMERIK-Software eingerichtet werden und durch Installation und Projektierung sowie in der späteren Anwendung nicht verändert werden. Ein Update oder Upgrade des Systems wird typischerweise in diesen Verzeichnissen erfolgen. Die Archivierung dieser Verzeichnisse ist deshalb nur als Rollback-Sicherung im Hintergrund erforderlich.

Die weitere Unterteilung dieser Verzeichnisse in SINUMERIK NCK / PLC / HMI und SINAMICS deckt die oben beschriebenen Datenbereiche ab und findet sich in allen genannten Verzeichnissen (Datenklassen) wieder.

CompactFlash Card	Datenbereich
SINUMERIK	NCK
	PLC
	HMI
SINAMICS	Antriebe

Struktur der CompactFlash Card

Das Dateisystem besteht auf oberster Ebene aus den Verzeichnissen "addon", "individual", "oem", "siemens" und "user": Diese Verzeichnisse besitzen grundsätzlich eine identische Struktur.

Nachfolgend ist der für die Konfiguration relevante Ausschnitt der Verzeichnisstruktur abgebildet:

Verzeichnis siemens				
/siemens/sinumerik				
	/hmi			
		/appl		Applikationen (Bedienbereiche)
		/base		Basissystemkomponenten
		/cfg		Alle Konfigurationsdateien
		/data		Versionsdaten
		/hlp		Online-Hilfedateien
		/hlps		Online-Hilfedateien gepackt und Versionsdateien
		/ico		Symboldateien
			/ico640	Symbole in der Auflösung 640x480
			/ico800	Symbole in der Auflösung 800x600
			/ico1024	Symbole in der Auflösung 1024x768
			/ico1280	Symbole in der Auflösung 1280x1024
			/ico1600	Symbole in der Auflösung 1600x1240
		/lng		Textdateien
		/lngs		Textdateien gepackt und Versionsdateien
		/osal		
			/ace	ACE/TAO
			/qt	Qt
		/proj		Easy Screen Projektierungen
		/template		Diverse Vorlagen
			/cfg	Vorlagen für Konfigurationsdateien
			/ing	Vorlagen für Textdateien
		/tmpp		Ablage temporärer Daten

Verzeichnis addon				
/addon/sinumerik				
	/hmi			
		/appl		Applikationen (Bedienbereiche)
		/cfg		Konfigurationsdateien
		/data		Versionsdaten
		/hlp		Online-Hilfedateien gepackt und Versionsdateien
		/ico		Symboldateien
			/ico640	Symbole in der Auflösung 640x480
			/ico800	Symbole in der Auflösung 800x600
			/ico1024	Symbole in der Auflösung 1024x768
			/ico1280	Symbole in der Auflösung 1280x1024
			/ico1600	Symbole in der Auflösung 1600x1240

Verzeichnis addon		
	/lng	Textdateien
	/lngs	Textdateien gepackt und Versionsdateien
	/proj	Easy Screen Projektierungen
	/template	Verschiedene Vorlagen

Verzeichnis oem				
/oem/sinumerik				
	/data			Versionsdaten
		/archive		Herstellerarchive
	/hmi			
		/appl		Applikationen (Bedienbereiche)
		/cfg		Konfigurationsdateien
		/data		Versionsdaten
		/hlp		Online-Hilfedateien
		/hlps		Online-Hilfedateien gepackt und Versionsdateien
		/ico		Symboldateien
			/ico640	Symbole in der Auflösung 640x480
			/ico800	Symbole in der Auflösung 800x600
			/ico1024	Symbole in der Auflösung 1024x768
			/ico1280	Symbole in der Auflösung 1280x1024
			/ico1600	Symbole in der Auflösung 1600x1240
		/lng		Textdateien
		/lngs		Textdateien gepackt und Versionsdateien
		/proj		Easy Screen Projektierungen
		/template		Verschiedene Vorlagen

Verzeichnis user				
/user/sinumerik				
	/data			Versionsdaten
		/archive		Anwenderspezifische Archive
		/prog		Anwenderspezifische Programme
	/hmi			
		/cfg		Konfigurationsdateien
		/data		Versionsdaten
		/hlp		Online-Hilfedateien
		/ico		Symboldateien
			/ico640	Symbole in der Auflösung 640x480
			/ico800	Symbole in der Auflösung 800x600
			/ico1024	Symbole in der Auflösung 1024x768

Verzeichnis user			
		/ico1280	Symbole in der Auflösung 1280x1024
		/ico1600	Symbole in der Auflösung 1600x1240
	/lng		Textdateien
	/log		Protokolldateien
	/md		Maschinendaten-Ansichten
	/proj		Easy Screen Projektierungen

3 Einstellungen der Bediensoftware

3.1 Zugriffsstufen

Zugriff auf Funktionen und Maschinendaten

Das Zugriffskonzept regelt den Zugriff auf Funktionen und Datenbereiche. Es gibt die Zugriffsstufen 0 bis 7, wobei 0 die höchste und 7 die niedrigste Stufe darstellt. Die Zugriffsstufen 0 bis 3 sind über Kennwort und 4 bis 7 über Schlüsselschalter-Stellungen verriegelt.

Zugriffsstufe	Verriegelt durch	Bereich	Datenklasse
0	---	(reserviert)	---
1	Kennwort: SUNRISE	Hersteller	Manufacturer (M)
2	Kennwort: EVENING	Service	Individual (I)
3	Kennwort: CUSTOMER	Anwender	User (U)
4	Schlüsselschalter Stellung 3	Programmierer, Einrichter	User (U)
5	Schlüsselschalter Stellung 2	qualifizierter Bediener	User (U)
6	Schlüsselschalter Stellung 1	ausgebildeter Bediener	User (U)
7	Schlüsselschalter Stellung 0	angelernter Bediener	User (U)

Das Kennwort bleibt solange gesetzt, bis es mit dem Softkey "Kennwort löschen" zurückgesetzt wird. Die Kennwörter können nach der Aktivierung geändert werden.

Sind z. B. die Kennwörter nicht mehr bekannt, so muss eine Neuinitialisierung (Hochlauf mit "NCK default data") durchgeführt werden. Dabei werden alle Kennwörter wieder auf die Voreinstellung (siehe Tabelle) zurückgesetzt. POWER ON setzt das Kennwort nicht zurück.

Schlüsselschalter

Die Zugriffsstufen 4 bis 7 erfordern eine entsprechende Schlüsselschalterstellung an der Maschinensteuertafel. Es gibt deshalb drei verschiedenfarbige Schlüssel. Jeder Schlüssel kann nur bestimmte Bereiche freischalten.

Bedeutung der Schlüsselschalterstellungen:

Zugriffsstufe	Schalterstellung	Schlüsselfarbe
4-7	0 bis 3	rot
5-7	0 bis 2	grün
6-7	0 und 1	schwarz
7	0 = Abziehstellung	kein Schlüssel gesteckt

Die Schlüsselschalterstellung muss immer vom PLC-Anwenderprogramm bearbeitet und entsprechend an die Nahtstelle gelegt werden.

Siehe auch

Das Kennwort kann auch über die PLC gelöscht werden (PI Dienste: PI LOGOUT).

Funktionshandbuch Grundfunktionen, Kapitel "PLC für SINUMERIK 828D" (P4)

3.2 So setzen und ändern Sie das Kennwort

Kennwort setzen

Um die Zugriffsstufe zu wechseln, wählen Sie den Bedienbereich "Inbetriebnahme":

1. Drücken Sie den Softkey "Kennwort".
2. Drücken Sie den Softkey "Kennwort setzen", um folgenden Dialog zu öffnen:

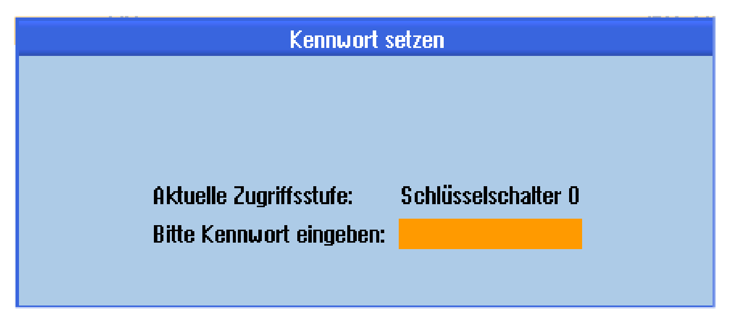


Bild 3-1 Kennwort setzen

3. Geben Sie ein Kennwort ein und bestätigen diese Eingabe mit "OK" oder mit der Taste <Input>.

   Ein gültiges Kennwort wird als gesetzt quittiert und die aktuell gültige Zugriffsstufe wird angezeigt. Ungültige Kennwörter werden abgewiesen.
4. Bevor Sie ein Kennwort für eine niedrigere Zugriffsstufe als die aktuelle aktiv setzen können, müssen Sie das Kennwort löschen.

   Durch Drücken des Softkeys "Kennwort löschen" wird das zuletzt gültige Kennwort gelöscht. Danach gilt die aktuelle Einstellung des Schlüsselschalters.

Kennwort ändern

Um das Kennwort zu ändern:

1. Drücken Sie den Softkey "Kennwort ändern", um folgenden Dialog zu öffnen:

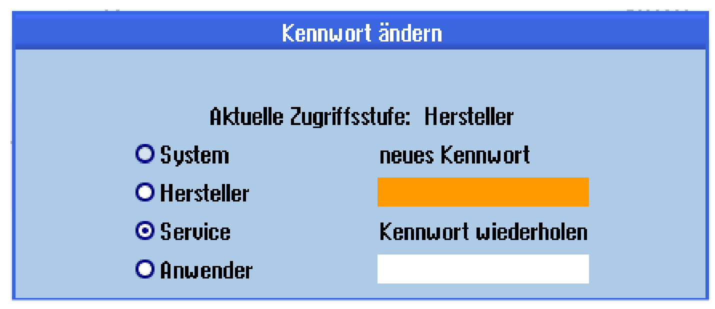


Bild 3-2 Kennwort ändern

2. Tragen Sie das neue Kennwort in beiden Eingabefeldern ein und bestätigen Sie anschließend mit den Softkey "OK". Wenn beide Kennwörter übereinstimmen, wird das neue Kennwort gültig und vom System übernommen.

3.3 So stellen Sie Datum und Uhrzeit ein

Voraussetzung

Hier sind Änderungen nur mit entsprechender Zugriffsberechtigung (ab "Anwender" und höher) möglich.

Datum und Uhrzeit einstellen

Vorgehensweise:

1. Wählen Sie den Bedienbereich "Inbetriebnahme" an.
2. Drücken Sie den Softkey "HMI".
3. Drücken Sie den Softkey "Datum Uhrzeit".

   Das Fenster "Datum/Uhrzeit" wird geöffnet.
4. Wählen Sie im Feld "Format" die gewünschten Formate für Datum- und Zeitanzeige aus.
5. Bestätigen Sie die Eingabe mit dem Softkey "OK".

   Die neuen Datum- und Zeitangaben werden übernommen und in der ersten Zeile in den Feldern "aktuell" ausgegeben.

Hinweis

Synchronisation der Uhrzeit

Die SINAMICS Antriebe haben keine Echtzeituhr. Die Uhrzeit und das Datum der SINAMICS Uhr wird im 10 Sekunden-Takt mit der SINUMERIK Echtzeituhr synchronisiert.

Das führt dazu, dass nach einer Änderung des Datums und/oder der Uhrzeit der SINUMERIK Echtzeituhr bis zu 10 Sekunden vergehen können, bis diese Änderung mit den SINAMICS Antrieben synchronisiert ist.

Wenn in dieser bis zu 10 Sekunden dauernden Zeitspanne SINAMICS Alarme auftreten (Alarmmummern 200000 -299999), erhalten diese SINAMICS Alarme noch den veralteten Datum-/Uhrzeitstempel. Die als Folge der SINAMICS-Alarme ausgelösten SINUMERIK-Alarme (Alarmnummer < 200000 und > 300000) hingegen erhalten bereits den neuen Datums-/Uhrzeitstempel.

3.4 Sprachen der Bediensoftware einstellen

3.4.1 Verfügbare Systemsprachen und Spracherweiterungen

Systemsprachen

Die SINUMERIK 828D wird im Grundumfang mit folgenden Systemsprachen ausgeliefert:

- Englisch
- Französisch
- Deutsch
- Italienisch
- Koreanisch
- Portugiesisch (Brasilien)
- Chinese simplified
- Spanisch
- Chinese traditional

Alle Systemsprachen sind im Auslieferungszustand der SINUMERIK 828D installiert, so dass eine Änderung der Sprache direkt über die Bedienoberfläche, ohne Nachladen von Systemsprachdaten erfolgen kann.

Spracherweiterungen

Weitere Systemsprachen sind auf der DVD Spracherweiterungen verfügbar. Eine Installationsanleitung ist auf der DVD vorhanden.

3.4.2 Eingabe asiatischer Schriftzeichen mit dem Input Method Editor

Input Method Editor (IME)

Im Programmeditor und im PLC-Alarmtexteditor ist das Editieren mit asiatischen Schriftzeichen möglich. Zur Unterstützung steht der Input Method Editor für folgende asiatische Sprachen zur Verfügung:

- Chinesisch vereinfacht
- Chinesisch traditionell (Taiwanesisch)
- Koreanisch

Mit der Tastenkombination <Alt>+<S> wird der Editor gestartet.

Editor mit aktiver Lernfunktion

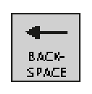
Für die Sprachen Chinesisch vereinfacht und Chinesisch traditionell bietet das System die Möglichkeit, mit Wörterbüchern zu arbeiten:

- ein Wörterbuch an der Steuerung bearbeiten
- ein Wörterbuch in die Steuerung importieren

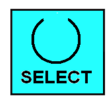
Wird eine Lautschrift eingegeben, zu denen keine Entsprechung in der Steuerung gespeichert ist, öffnet der Editor eine Lernfunktion. Diese Funktion ermöglicht das Zusammenstellen von Silben oder Wörtern, die nach dem Speichern dauerhaft zur Verfügung stehen. Der Editor zeigt die Zusammenstellung der chinesischen Zeichen neben der Pinyin-Lautschrift an. Nach dem vollständigen Zusammenstellen, ist das Wort mit der <Input>-Taste zu speichern. Es wird gleichzeitig in das zugehörige Eingabefeld eingefügt.

Wörterbücher bearbeiten

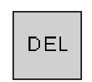
Wird diese Funktion aktiviert, erscheint eine weitere Zeile, in der die zusammengesetzten Schriftzeichen und Laute angezeigt werden. Der Editor bietet zu diesem Laut verschiedene Schriftzeichen an, aus denen ein Zeichen durch die Eingabe der entsprechenden Ziffer (1 ... 9) ausgewählt werden kann. Der Cursor zur Eingabe kann mittels Taste <TAB> zwischen dem Feld zusammengesetzte Laute und Lauteingabe wechseln.



Ist der Cursor auf das obere Feld gesetzt, kann der Bediener die angezeigte Zusammensetzung mit der Taste <Backspace> rückgängig machen.



Mit der Taste <Select> erfolgt das Speichern der angezeigten Schriftzeichen.



Die Taste <Delete> löscht die gezeigte Zeichengruppe aus dem Wörterbuch.

Wörterbücher importieren

Ein Wörterbuch kann mit jedem Unicode Editor erstellt werden, indem mit der Pinyin-Lautschrift die entsprechenden chinesischen Zeichen anfügt werden. Beinhaltet die Lautschrift mehrere chinesische Zeichen, darf die Zeile keine weitere Entsprechung enthalten. Falls mehrere Entsprechungen zu einer Lautschrift existieren, sind diese zeilenweise im Wörterbuch anzugeben. Ansonsten können mehrere Zeichen pro Zeile angegeben werden.

Die erstellte Datei ist im UTF8-Format unter dem Namen chs_user.txt (vereinfachtes Chinesisch) oder cht_user.txt (traditionelles Chinesisch) zu speichern.

Zeilenaufbau:

Pinyin Lautschrift <TAB> chinesische Zeichen <LF>

ODER

Pinyin Lautschrift <TAB> chinesisches Zeichen1<TAB> chinesisches Zeichen2 <TAB> … <LF>

<TAB> - Tabulator

<LF> - Zeilenumbruch

Legen Sie das erstellte Wörterbuch in einem der folgenden Pfade ab:

`../user/sinumerik/hmi/ime/`

`../oem/sinumerik/hmi/ime/`

Mit dem nächsten Aufruf des chinesischen Editors fügt dieser den Inhalt des Wörterbuchs in das Systemwörterbuch ein.

Beispiel:

```
ai          哎  哀  唉  埃  挨
caise       彩色
hongse      紅色
huise       灰色
heli        河裏
zuihaowan   最好玩
```

Hinweis

Die Wörterbücher werden auch im Inbetriebnahmearchiv gesichert, wenn "HMI-Daten" angewählt ist (siehe auch: Daten sichern in Archiven (Seite 390))

3.4.3 So geben Sie chinesische Schriftzeichen ein

Chinesische Zeichen eingeben

Das Auswählen eines Zeichens erfolgt mittels Lautsprache ("pinyin method"): dessen Laut sich durch das Zusammensetzen von lateinischen Buchstaben bilden lässt.

1. Geben Sie den Laut durch Zusammensetzen von lateinischen Buchstaben ein.

   Der Editor bietet eine Auswahl von Zeichen an, die diesem Laut entsprechen.

2. Wählen Sie das gewünschte Zeichen mit den Cursor-Tatsen aus:

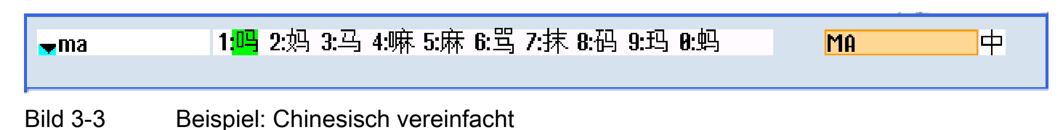


Bild 3-3 Beispiel: Chinesisch vereinfacht

Schaltet man das Optionsfeld mit der Taste <SELECT> auf die Eingabe lateinischer Buchstaben um, werden die Eingaben direkt an das Eingabefeld weitergeleitet, das vor dem Öffnen des chinesischen Editors den Focus hatte.

Zhuin Eingabe bei Chinesisch Traditionell

Zusätzlich steht bei Chinesisch Traditionell noch folgende Eingabemöglichkeit zur Verfügung:

1. Zum Bilden der einzelnen Silben verwenden Sie den Ziffernblock der Tastatur.

   Jeder Ziffer sind eine Anzahl von Buchstaben zugeordnet, die durch das ein- oder mehrmalige Betätigen der Ziffertaste ausgewählt werden können.

2. Um die getroffene Auswahl, die im Zhuyin-Eingabefeld angezeigt wird, zu übernehmen, bestätigen Sie mit der Taste <INPUT> oder der Eingabe einer weiteren Ziffer.



Bild 3-4 Beispiel: Zhuin Methode

3.4.4 So geben Sie koreanische Schriftzeichen ein

Zeichen mit Hilfe einer Matrix eingeben

Steht nur die Tastatur an der Steuerung zur Verfügung, kommt ein Matrixverfahren zur Anwendung, das nur den Ziffernblock benötigt:

1. Mit der ersten Zahl wählen Sie die Zeile aus: die Zeile wird farblich hervorgehoben.
2. Mit der zweiten Zahl wählen Sie die Spalte aus: das Zeichen wird kurzzeitig farblich hervorgehoben und in das Feld "Schriftzeichen" übernommen.

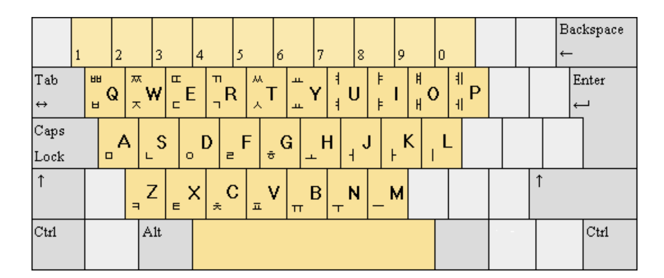
Mit der Taste <SELECT> schalten Sie zwischen koreanisch und lateinisch um.



Bild 3-5 Beispiel: Programmeditor

Koreanische Tastatur verwenden

Zur Eingabe von koreanischen Schriftzeichen benötigt der Bediener eine Tastatur mit der unten dargestellten Tastaturbelegung. Diese Tastatur entspricht bezüglich der Tastenbelegung einer englischen QWERTY- Tastatur, wobei die erhaltenen Events in Silben zusammengefasst werden müssen.



Beispiel: Koreanische Tastatur

Das Alphabet (Hangeul) besteht aus 24 Buchstaben: 14 Mitlaute und 10 Selbstlaute. Die Silbenbildung erfolgt durch das Zusammensetzten der Mit- und Selbstlaute.

3.5 Lizenzen prüfen und eingeben

Nutzung

Die Nutzung der auf einer SINUMERIK Steuerung installierten Systemsoftware und der aktivierten Optionen erfordert, dass die dafür erworbenen Lizenzen der Hardware zugeordnet werden. Im Rahmen dieser Zuordnung wird aus den Lizenznummern der Systemsoftware und der Optionen sowie der Hardware-Seriennummer ein License Key generiert. Hierbei wird über das Internet auf eine von Siemens administrierte Lizenzdatenbank zugegriffen. Abschließend werden die Lizenzinformationen einschließlich License Key auf die Hardware übertragen.

Der Zugriff auf die Lizenzdatenbank erfolgt mit dem Web License Manager.

Web License Manager

Mit dem Web License Manager kann die Zuordnung von Lizenzen zur Hardware in einem Standard Web Browser vorgenommen werden. Zum Abschluss des Zuordnungsvorganges muss der License Key an der Steuerung manuell über die Bedienoberfläche eingegeben werden.

Internetadresse: Web License Manager (http://www.siemens.com/automation/license)

Hinweis

SINUMERIK Softwareprodukte

Ist für ein SINUMERIK Softwareprodukt kein License Key aktiviert oder vorhanden, wird von der Steuerung der Alarm 8081 ausgegeben und NC-START kann nicht ausgeführt werden.

Siehe auch

Definitionen zum Lizenz-Management (Seite 55)

3.5.1 So geben Sie einen Lizenz-Schlüssel ein

Voraussetzung

Für die Nutzung der aktivierten Optionen benötigen Sie die entsprechenden Lizenzen. Nach der Lizenzierung der Optionen im Web License Managers erhalten Sie einen "License Key", der alle lizenzpflichtigen Optionen enthält und nur für Ihre System CompactFlash Card gültig ist.

Um Optionen zu setzen oder zurückzusetzen, werden die Zugriffsrechte ab "Hersteller" benötigt.

License Key einlesen oder eingeben

Vorgehensweise:

1. Wählen Sie den Bedienbereich "Inbetriebnahme" an.
2. Drücken Sie die Menüfortschalt-Taste.
3. Drücken Sie den Softkey "Lizenzen".
4. Das Fenster "Lizenzierung" wird geöffnet
5. Markieren Sie die Eingabezeile, um einen neuen License Key einzugeben.

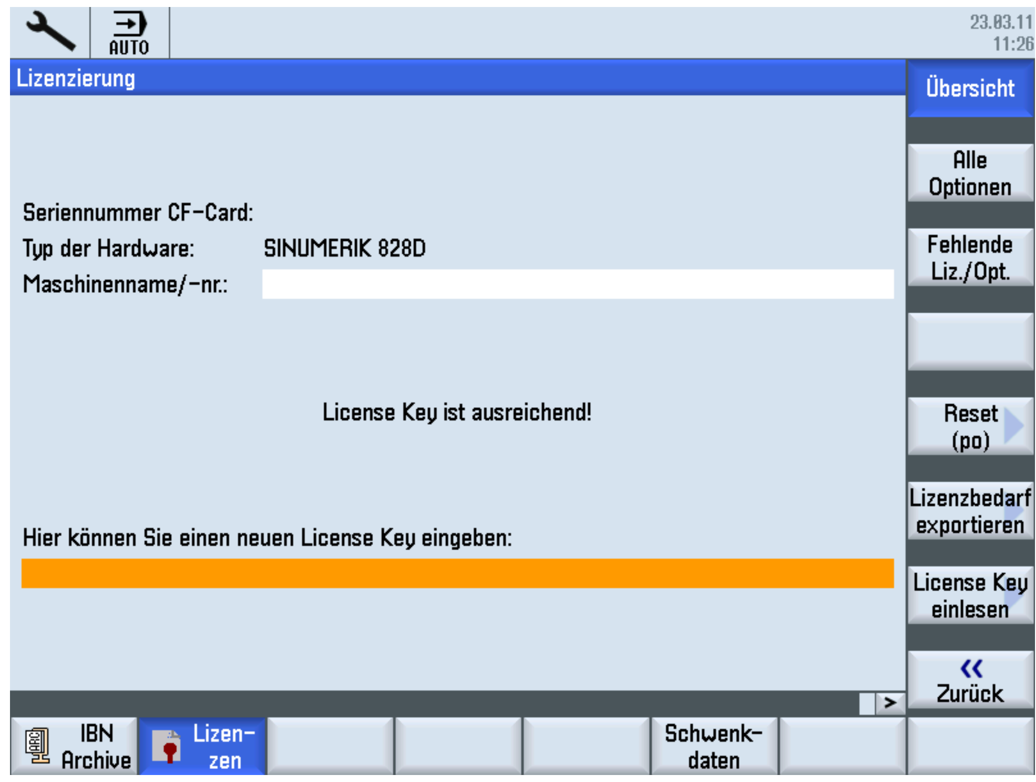


Bild 3-6 Lizenzierung

Weitere Aktionen sind:

- Softkey "Alle Optionen": Alle Optionen werden angezeigt.
- Softkey "Fehlende Lizenzen/Optionen": Fehlende Lizenzen und Optionen werden angezeigt: So ermitteln Sie fehlende Lizenzen/Optionen (Seite 54)
- Softkey: "Lizenzbedarf exportieren": Die fehlenden Lizenzen werden in eine Datei exportiert und können auf einem Speichermedium gesichert werden.
- Softkey: "License Key einlesen ": Der License Key wird aus der Lizenz-Datei eingelesen.

  Diese Datei wird mitgeliefert oder kann über den Web License Manager (http://www.siemens.com/automation/license) bezogen werden.

3.5.2 So ermitteln Sie fehlende Lizenzen/Optionen

Lizenzbedarf ermitteln

Vorgehensweise:

1. Drücken Sie den Softkey "Alle Optionen", um die gesamten Optionen aufzulisten, die für diese Steuerung anwählbar sind.
2. Aktivieren oder deaktivieren Sie die notwendigen Optionen in der Spalte "gesetzt":
   – Kontrollkästchen markieren
   – Eingabe der Anzahl von Optionen
3. Drücken Sie den Softkey "Fehlende Liz./Opt.", um alle Optionen anzuzeigen, die lizenziert sind. In der Spalte "gesetzt" können Sie Optionen, die Sie nicht benötigen, wieder abwählen.

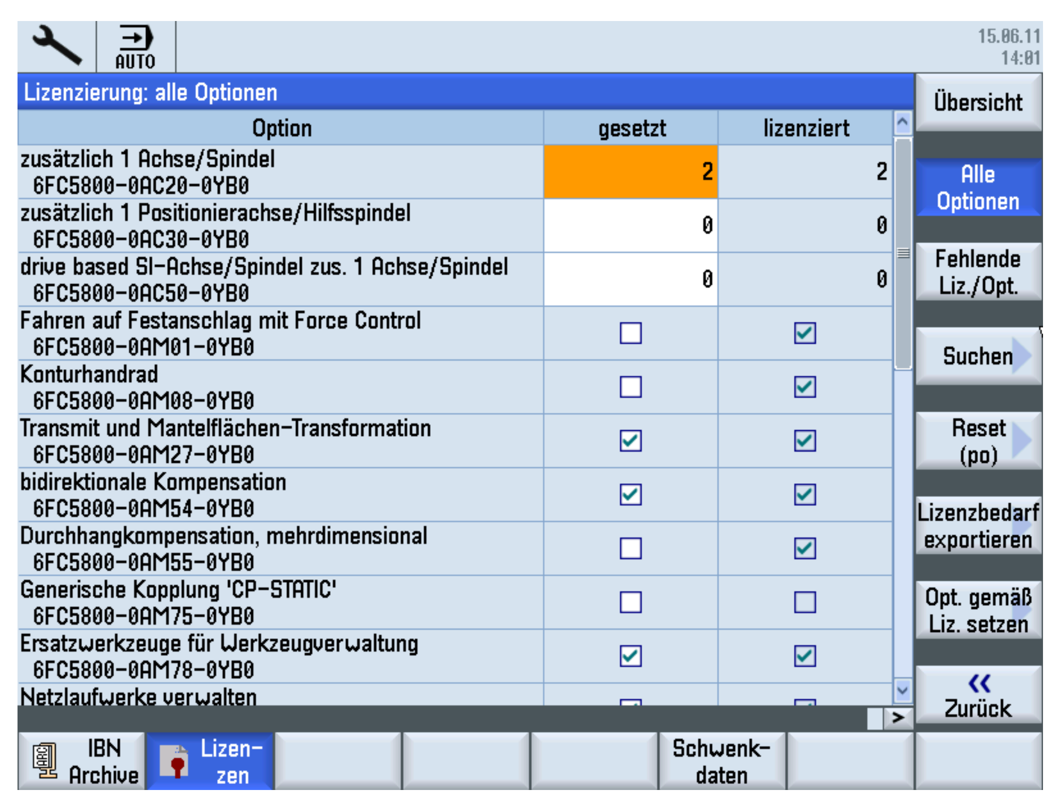


Bild 3-7 Lizenzierung (Beispiel)

4. Drücken Sie den Softkey "Opt. gemäß Liz. setzen", um alle im License Key enthaltenen Optionen zu aktivieren. Die nachfolgende Sicherheitsabfrage bestätigen Sie mit "OK".
5. Um neu aktivierte Optionen wirksam zu setzen, drücken Sie den Softkey "Reset (po)". Sie erhalten eine Sicherheitsabfrage.
6. Drücken Sie den Softkey "OK", um einen Warmstart durchzuführen.

   - ODER -

7. Drücken Sie den Softkey "Abbruch", um den Vorgang abzubrechen.

3.5.3 Definitionen zum Lizenz-Management

Produkt

Ein Produkt ist im Rahmen des Lizenzmanagements von SINUMERIK → Softwareprodukten durch folgende Daten gekennzeichnet:

- Produktbezeichnung
- Bestellnummer
- → Lizenznummer

Softwareprodukt

Als Softwareprodukt wird allgemein ein Produkt bezeichnet, das auf einer → Hardware zur Bearbeitung von Daten installiert wird. Im Rahmen des Lizenzmanagements von SINUMERIK Softwareprodukten wird für die Nutzung jedes Softwareproduktes eine entsprechende → Lizenz benötigt.

Certificate of License (CoL)

Das CoL ist der Nachweis der → Lizenz. Das Produkt darf nur durch den Inhaber der → Lizenz oder beauftragten Personen genutzt werden. Auf dem CoL befinden sich unter anderem folgende für das Lizenzmanagement relevante Daten:

- Produktname
- → Lizenznummer
- Lieferscheinnummer
- → Hardware-Seriennummer

Hardware

Als Hardware im Rahmen des Lizenzmanagements von SINUMERIK → Softwareprodukten wird die Komponente einer SINUMERIK Steuerung bezeichnet, der aufgrund ihrer eindeutigen Kennung → Lizenzen zugeordnet werden. Auf dieser Komponente werden auch die Lizenzinformationen remanent gespeichert, z. B. auf einer → CompactFlash Card.

CompactFlash Card

Die CompactFlash Card repräsentiert als Träger aller remanenten Daten einer SINUMERIK solution line Steuerung die Identität dieser Steuerung. Die CompactFlash Card ist eine Speicherkarte, die in die → Control Unit von außen steckbar ist. Auf der CompactFlash Card befinden sich folgende für das Lizenzmanagement relevante Daten:

- → Hardware-Seriennummer
- Lizenzinformationen einschließlich → License Key

Hardware-Seriennummer

Die Hardware-Seriennummer ist unveränderlicher Bestandteil der → CompactFlash Card. Über sie wird eine Steuerung eindeutig identifiziert. Die Hardware-Seriennummer kann ermittelt werden über:

- → Certificate of License
- Bedienoberfläche
- Aufdruck auf der → CompactFlash Card

Lizenz

Eine Lizenz wird als Recht zur Nutzung eines → Softwareproduktes vergeben. Die Repräsentanten dieses Rechtes sind:

- → Certificate of License (CoL)
- → License Key

Lizenznummer

Die Lizenznummer ist das Merkmal einer → Lizenz, über das sie eindeutig identifiziert wird.

License Key

Der License Key ist der "technische Repräsentant" der Summe aller → Lizenzen, die einer bestimmten, durch ihre → Hardware-Seriennummer eindeutig gekennzeichneten → Hardware, zugeordnet sind.

Option

Eine Option ist ein SINUMERIK → Softwareprodukt, dass nicht in der Grundausführung enthalten ist und für dessen Nutzung eine → Lizenz erworben werden muss.

3.6 PLC Anwender-Alarme projektieren

Anwender PLC-Alarme erstellen

Die PLC-Alarme im Bereich von **700 000 - 700 247** werden vom Maschinenhersteller projektiert. Es wird die Zugriffstufe "Hersteller" mit entsprechendem Kennwort vorausgesetzt.

Um Anwender PLC-Alarme über die Bedienoberfläche einzugeben, wählen Sie im Bedienbereich "Inbetriebnahme" → "HMI"→ "Alarmtexte".

Dann erhalten Sie folgende Auswahl:

Alarmtexte für	Bezeichnung der xml-Datei
Anwender Zyklen-Alarme	oem_alarms_cycles
Anwender PLC-Alarme	oem_alarms_plc
Anwender Teileprogramm-Meldetexte	oem_partprogram_messages

Laden der Anwender PLC-Alarme

Das Laden der Alarmtextdateien erfolgt nur beim Hochlauf.

- Attribut "Alarm": rot, wird in der "Alarmliste" angezeigt.
- Attribut "Meldung": schwarz, wird unter "Meldungen" angezeigt.

Um die Alarmtexte zu laden, führen Sie einen Neustart des Systems durch.

Literatur

Beschreibung der Alarme mit Systemreaktionen und Löschkriterien in: SINUMERIK 828D Diagnosehandbuch

3.6.1 Struktur der Anwender PLC-Alarme

Struktur eines Anwender PLC-Alarms

Die Anwender PLC-Alarme haben folgende Struktur:

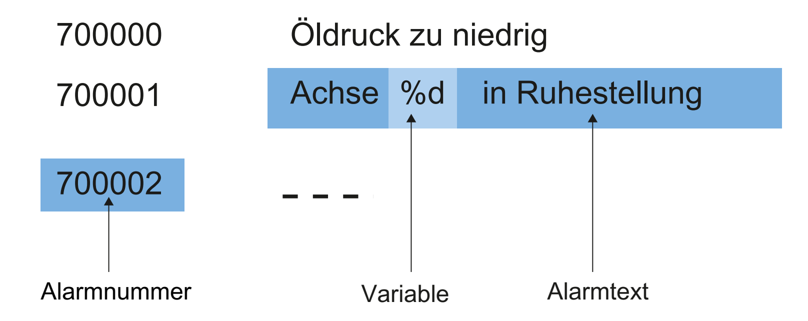


Bild 3-8 Alarmstruktur

Die folgende Tabelle zeigt die Wirkungsweise der PLC-Alarme:

1. Durch das PLC-Signal wird der Alarm mit der zugehörigen Nummer ausgelöst und ausgegeben.
2. Falls zu diesem Alarm eine Variable projektiert wurde, steht der Wert dieser Variablen im angegebenen Datenwort der PLC-Variable.
3. Im Index des MD14516[x] wird die Reaktion des NCK beim Auslösen des Alarms definiert (siehe Tabelle weiter unten).
4. Der Alarmtext ist frei wählbar und darf bis zu 255 Zeichen lang sein.

Alarm Nummer	PLC Signal	PLC Variable	Alarm Reaktion (MD)	Alarmtext
700000	DB1600.DBX0.0	DB1600.DBD1000	14516[0]	Alarm 1
700001	DB1600.DBX0.1	DB1600.DBD1004	14516[1]	Alarm 2
700002	DB1600.DBX0.2	DB1600.DBD1008	14516[2]	Alarm 3
700003	DB1600.DBX0.3	DB1600.DBD1012	14516[3]	Alarm 4
700004	DB1600.DBX0.4	DB1600.DBD1016	14516[4]	Alarm 5
700005	DB1600.DBX0.5	DB1600.DBD1020	14516[5]	Alarm 6
700006	DB1600.DBX0.6	DB1600.DBD1024	14516[6]	Alarm 7

Fortsetzung:				
700247	DB1600.DBX30.7	DB1600.DBD1988	14516[247]	Alarm 248

Reaktion des NCK definieren

Folgende Reaktionen des NCK sind möglich:

MD14516[x]	Bedeutung
Bit 0	NC Start gesperrt
Bit 1	Einlesesperre
Bit 2	Vorschubsperre für alle Achsen
Bit 3	NOT AUS
Bit 4	PLC in Stopp
Bit 5	reserviert
Bit 6	Definition für Alarm oder Meldung Bit 6=1: → Alarm, Bit 6=0: → Meldung
Bit 7	POWER ON

Alarmtexte mit Variablen projektieren

Folgende Datentypen sind für Variablen im Alarmtext zulässig:

Variable	Bedeutung
%b	binäre Darstellung eines 32 Bit Werts
%d	ganzzahlige Dezimalzahl
%f	4 Byte Gleitkommazahl
%i	ganzzahlige Dezimalzahl mit Vorzeichen
%o	ganzzahlige Oktalzahl
%u	Dezimalzahl ohne Vorzeichen
%x	ganzzahlige Hexadezimalzahl

3.6.2 So erstellen Sie Anwender PLC-Alarme

Hinweise für die Bearbeitung

Für die Bearbeitung der Dateien sind folgende Hinweise zu beachten:

- Die Dateien sind extern auf PG/PC mit einem Texteditor (z. B. Notepad) oder mit einem xml-Editor zu bearbeiten. Die Struktur darf nicht geändert werden.
- Die erstellten Alarmtextdateien werden auf der CompactFlash Card in folgendes Verzeichnis kopiert: `oem/sinumerik/hmi/lng`
- Damit die Alarmtextdatei vom System erkannt wird, muss der Dateiname in Kleinbuchstaben geschrieben sein.
- Die Alarmtextdatei wird beim Hochlauf des Systems konvertiert: Damit die Alarme aktiviert werden, ist ein Neustart des Systems nötig.

Vorgehensweise

Um eine größere Anzahl von Alarmen zu editieren, erstellen Sie zunächst 2 oder 3 Alarme direkt an der Steuerung. Damit wird die Datei oem_alarms_plc_xxx.ts erzeugt und Sie haben eine "Dokumentvorlage" mit der korrekten Struktur, die Sie dann mit weiteren Alarmen ergänzen. Das Kürzel "xxx" steht für die Sprache, in der die Datei erstellt wird.

1. Wählen Sie den Bedienbereich "Inbetriebnahme".
2. Drücken Sie den Softkey "HMI".
3. Drücken Sie den Softkey "Alarmtexte": Das Fenster "Datei auswählen" wird geöffnet.
4. Wählen Sie "oem_alarms_plc", um Anwender PLC-Alarmtexte zu erstellen.
5. Geben Sie im Feld "Nummer" die Alarmnummer und im Feld "Text" den gewünschten Alarmtext ein. Die Alarmnummern und zugehörige Alarmtexte müssen nicht fortlaufend sein. Wird ein Alarm ohne projektierten Text ausgelöst, wird nur die Alarmnummer ausgegeben.

Suchen innerhalb der Alarmtexte

Um einen Text oder eine Zeichenfolge zu suchen:

1. Drücken Sie den Softkey "Suchen >". Das Fenster "Suchen" wird geöffnet; und ein neues Menü auf der vertikalen Softkey-Leiste wird angezeigt.
2. Geben Sie im Feld "Text" den zu suchenden Begriff ein.
3. Positionieren Sie den Cursor in das Feld "Richtung" und wählen Sie mit der Taste <SELECT> die Suchrichtung (vorwärts, rückwärts).
4. Aktivieren Sie das Kontrollkästchen "Groß-Kleinschreibung unterscheiden", wenn bei dem eingegebenen Text zwischen Groß-Kleinschreibung unterschieden werden muss.
5. Drücken Sie den Softkey "Suchen + Ersetzen". Analog wird das Fenster "Suchen und Ersetzen" geöffnet.
6. Drücken Sie den Softkey "OK", um die Suche zu starten.
7. Drücken Sie den Softkey "Abbruch", um die Suche abzubrechen.

Weitere Navigationsmöglichkeiten sind:

- Softkey "Gehe zum Anfang":

  Der Cursor springt auf den ersten Eintrag der ausgewählten Alarmtextdatei
- Softkey "Gehe zum Ende":

  Der Cursor springt auf den letzten Eintrag der ausgewählten Alarmtextdatei.

Siehe auch

Liste der Sprachkennzeichen für Dateinamen (Seite 403)

Beispiel: So erstellen Sie eine Online-Hilfe für Anwender PLC-Alarme (Seite 75)

3.6.3 Alarm-Protokoll konfigurieren

Protokollierung

Das Alarmprotokoll konfigurieren Sie im Bedienbereich "Diagnose".

Alle Alarme und Meldungen mit ihren Kommen- und Gehen-Zeitstempeln werden in chronologischer Reihenfolge protokolliert. Ausgenommen sind Meldungen des Typs "msg" vom NC-Teileprogramm. Alle zum Zeitpunkt der Anzeige des Protokolls nicht mehr aktiven Alarme oder Meldungen (sog. "historische Ereignisse") bleiben auch erhalten.

Das Alarmprotokoll ist als Ringpuffer organisiert (Voreinstellung). Die jeweils ältesten Einträge werden in folgenden Fällen durch neue Ereignisse überschrieben:

- wenn die maximale Größe überschritten wird (zulässiger Bereich: 0 - 32000).
- wenn die Ereignisse vor dem letzten Einschaltzeitpunkt des Systems liegen.

Dauerhafte Sicherung

Um das Alarmprotokoll dauerhaft zu sichern, wird das Alarmprotokoll auf die CompactFlash Card geschrieben.

Hinweis

Alarmprotokoll speichern

Zur dauerhaften Sicherung wird das Alarmprotokoll auf die CompactFlash Card geschrieben, die nur eine begrenzte Anzahl von Schreibzyklen ermöglicht.

- Stellen Sie sicher, dass die Sicherung nur in begründeten Fällen erfolgt!
- Stellen Sie sicher, dass Sie die Einstellung "bei jedem Ereignis" wieder rückgängig machen, sobald Sie die Speicherung des Alarmprotokolls nicht mehr benötigen.

Voreinstellung: Das Alarmprotokoll wird nicht gesichert.

Alarme filtern

Um die Anzahl der Ereignisse im Alarmprotokoll zu begrenzen, richten Sie ein Filter ein. Für folgende Filter wählen Sie den Bereich aus:

- Zeitintervall
- Alarmnummernband

3.6.4 So konfigurieren Sie die Protokollierung

Protokollierung konfigurieren

Vorgehensweise:

1. Wählen Sie den Bedienbereich "Diagnose" an.
2. Drücken Sie den Softkey "Alarmprotok."
3. Drücken Sie den Softkey "Einstellungen".
4. Geben Sie im Feld "Anzahl Einträge" die gewünschte Zahl ein, um die maximale Anzahl an Kommen- und Gehen-Ereignissen zu ändern.

   Voreinstellung sind 500 Ereignisse; zulässiger Wertebereich 0 - 32000.
5. Wählen Sie unter "Schreibmodus Datei" die Art der Protokollierung aus:
   – "**aus**", wenn die Ereignisse nicht in eine Datei geschrieben werden sollen.
   – "**bei jedem Ereignis**", wenn jedes Ereignis in die Datei geschrieben werden soll.
   – "**zeitgesteuert**", wenn die Datei nach einem bestimmten Zeitintervall neu geschrieben werden soll. Sie erhalten ein zusätzliches Eingabefeld "Schreibintervall", um die Zeit in Sekunden vorzugeben.
6. Drücken Sie den Softkey "Protokoll speichern", um das Alarmprotokoll zu speichern.

Die Einstellungen werden erst nach einem Neustart des Systems wirksam.

Konfigurationsdatei bearbeiten

Vorgehensweise:

1. Kopieren Sie die Konfigurationsdatei "oem_alarmprot_slaesvcconf.xml" aus dem Verzeichnis `/siemens/sinumerik/hmi/template/cfg`.
2. Fügen Sie die Datei in das Verzeichnis `/oem/sinumerik/hmi/cfg` oder `/user/sinumerik/hmi/cfg`
3. Geben Sie der Datei den Namen "slaesvcconf.xml"
4. Öffnen Sie die anwenderspezifische Datei "slaesvcconf.xml" im Editor.
5. Tragen Sie im Bezeichner <Records type ... /> die Anzahl der auszugebenden Ereignisse ein.

   Der voreingestellte Wert ist 500. Die zulässige Anzahl liegt im Bereich von 0 ... 32000.

ODER:

Die Anzahl der auszugebenden Ereignisse und die Art der Protokollierung können Sie auch direkt über die Bedienoberfläche eingeben:

1. Wählen Sie im Bedienbereich "Diagnose" den Softkey "Alarmprotokoll" → "Einstellungen >".

   Sobald hier Änderungen gegenüber der Voreinstellung vorgenommen werden, wird die Datei "slaesvcconf.xml" automatisch im Verzeichnis `/user/sinumerik/hmi/cfg` angelegt.

2. Im Bezeichner <DiskCare type="int" value="-1"/> tragen Sie den Modus der Persistenz-Sicherung ein. Folgende Werte sind möglich:

   -1: Es erfolgt keine Sicherung des Alarmprotokolls (Voreinstellung).

   0: Jedes Alarmereignis stößt eine sofortige Sicherung des Alarmprotokolls an.

   >0: Zeit für die Sicherung des Protokolls in Sekunden:

   Sofern sich eine Änderung ergeben hat, wird das Protokoll zeitgetriggert alle n > 0 Sekunden gesichert.

3. Im Bezeichner <Filter> passen Sie das Filter für die Art der Einträge an.

   Hierfür gilt:

   – Ein Alarm-Ereignis wird nur dann in das Protokoll aufgenommen, wenn es das Filterkriterium erfüllt.

   – Wenn mehrere Filter definiert werden, sind diese durch die logischen Operatoren OR oder AND zu verknüpfen.

Die Einstellungen werden erst nach einem Neustart des Systems wirksam.

Hinweis

Anzahl von Ereignissen

Jedes Kommen- und Gehen-Ereignis eines Alarms oder einer Meldung benötigt einen eigenen Eintrag, auch wenn sie zum selben Alarm oder zur selben Meldung gehören.

Weiterhin sind Quittungs-Ereignisse im Alarmprotokoll enthalten. Sie benötigen ebenfalls einen Eintrag, auch wenn diese in der Protokoll-Anzeige nicht erkennbar sind.

Beispiele

Es werden alle Alarme protokolliert, die folgende Bedingung erfüllen:

- CLEARINFO ≠ 15, d.h. ohne Teileprogramm-Meldungen:

```
<CONFIGURATION>
  <Protocol>
    <Filters>
      <Siemens_Filter_01 type="QString" value="CLEARINFO NOT 15" />
    </Filters>
  </Protocol>
</CONFIGURATION>
```

- "SEVERITY größer als 10" und "kleiner als 500" :

```
<CONFIGURATION>
  <Protocol>
    <Filters>
      <Filter_01 type="QString" value= "SEVERITY HIGHER 10
 AND SEVERTY LOWER 500" />
    </Filters>
  </Protocol>
</CONFIGURATION>
```

3.7 OEM-spezifische Online-Hilfe erstellen

Übersicht

Über die bereits im System vorhandene Online-Hilfe hinaus haben Sie die Möglichkeit, eine herstellerspezifische Online-Hilfe zu erstellen und in der Bedien-Software zu ergänzen.

Diese Online-Hilfe wird im HTML-Format erstellt, d. h. sie besteht aus untereinander verlinkten HTML-Dokumenten. Das gesuchte Thema wird in einem gesonderten Fenster über ein Inhalts- oder Stichwortverzeichnis aufgerufen. Ähnlich wie in einem Dokumenten-Browser (z. B. Windows Explorer) wird in der linken Fensterhälfte eine Auswahlübersicht angezeigt und wenn Sie das gewünschte Thema anklicken, wird die Erläuterung dazu in der rechten Fensterhälfte angezeigt.

Eine kontextsensitive Anwahl von Online-Hilfeseiten ist nicht möglich.

Genereller Ablauf:

1. HTML-Dateien erzeugen
2. Hilfebuch erzeugen
3. Online-Hilfe in Bedien-Software einbinden
4. Hilfedateien im Zielsystem ablegen

Weitere Anwendungsfälle

Es können Online-Hilfen zu folgenden OEM-spezifischen Erweiterungen erstellt und zum SINUMERIK Operate Online-Hilfe-System ergänzt werden:

- Online-Hilfe zu Zyklen und/oder M-Funktionen des Maschinenherstellers, die die Programmiermöglichkeiten der SINUMERIK-Steuerungen erweitern. Diese Online-Hilfe wird genauso aufgerufen SINUMERIK Operate Online-Hilfe "Programmieren".

  Siehe auch: Beispiel: So erstellen Sie eine Programmier-Online-Hilfe (Seite 80)

- Online-Hilfe zu OEM-spezifischen Variablen des Maschinenherstellers. Diese Online-Hilfe wird aus der Variablenansicht von SINUMERIK Operate aufgerufen.

  Siehe auch: Beispiel: So erstellen Sie eine Online-Hilfe für NC-/PLC-Variablen (Seite 78)

3.7.1 Aufbau und Syntax der Konfigurationsdatei

Syntax-Beschreibung der "slhlp.xml"

Um das Hilfebuch in das vorhandene Online-Hilfesystem der Bedienoberfläche einzubinden, benötigen Sie die Konfigurationsdatei "slhlp.xml":

<table>
<tr><th>Tag</th><th>Anzahl</th><th colspan="2">Bedeutung</th></tr>
<tr><td>CONFIGURATION</td><td>1</td><td colspan="2">Root Element des XML-Dokuments: Kennzeichnet, dass es sich um eine Konfigurationsdatei handelt.</td></tr>
<tr><td>OnlineHelpFiles</td><td>1</td><td colspan="2">Leitet die Sektion der Hilfebücher ein.</td></tr>
<tr><td><help_book></td><td>*</td><td colspan="2">Leitet die Sektion eines Hilfebuchs ein.</td></tr>
<tr><td rowspan="3">EntriesFile</td><td rowspan="3">1</td><td colspan="2">Dateiname des Hilfebuchs mit den Inhalts- und Stichworteinträgen.
Attribute:</td></tr>
<tr><td>value</td><td>Name der XML-Datei</td></tr>
<tr><td>type</td><td>Datentyp des Werts (QString)</td></tr>
<tr><td rowspan="3">III-Technology</td><td rowspan="3">0,1</td><td colspan="2">Gibt die Technologie an, für die das Hilfebuch gilt.
"All" gilt dabei für alle Technologien.
Wenn das Hilfebuch für mehrere Technologien gilt, werden die Technologien durch ein Komma getrennt angegeben.
Mögliche Werte:
All, Universal, Milling, Turning, Grinding, Stroking, Punching
Attribute:</td></tr>
<tr><td>value</td><td>Technologieangabe</td></tr>
<tr><td>type</td><td>Datentyp des Werts (QString)</td></tr>
<tr><td rowspan="3">DisableSearch</td><td rowspan="3">0,1</td><td colspan="2">Stichwortsuche für das Hilfebuch ausschalten.
Attribute:</td></tr>
<tr><td>value</td><td>true, false</td></tr>
<tr><td>type</td><td>type Datentyp des Werts (bool)</td></tr>
<tr><td rowspan="3">DisableFullTextSearch</td><td rowspan="3">0,1</td><td colspan="2">Volltextsuche für das Hilfe-Buch ausschalten.
Attribute:</td></tr>
<tr><td>value</td><td>true, false</td></tr>
<tr><td>type</td><td>type Datentyp des Werts (bool)</td></tr>
<tr><td rowspan="3">DisableIndex</td><td rowspan="3">0,1</td><td colspan="2">Stichwortverzeichnis für das Hilfebuch ausschalten.
Attribute:</td></tr>
<tr><td>value</td><td>true, false</td></tr>
<tr><td>type</td><td>type Datentyp des Werts (bool)</td></tr>
<tr><td rowspan="3">DisableContent</td><td rowspan="3">0,1</td><td colspan="2">Inhaltsverzeichnis für das Hilfebuch ausschalten.
Attribute:</td></tr>
<tr><td>value</td><td>true, false</td></tr>
<tr><td>type</td><td>type Datentyp des Werts (bool)</td></tr>
</table>

Tag	Anzahl	Bedeutung	
DefaultLanguage	0,1	Kürzel für die Sprache die angezeigt werden soll, wenn für das Hilfebuch die aktuelle Landessprache vorhanden ist. Attribute:	
		value	chs, cht, deu, eng, esp, fra, ita, kor, ptb ...
		type	Datentyp des Werts (QString)

Für die Spalte "Anzahl" gilt: * bedeutet 0 oder mehrere

Beispiel für eine Datei "slhlp.xml"

Im folgenden Beispiel wird das Hilfebuch "hmi_myhelp.xml" konfiguriert; das Stichwortverzeichnis ist nicht aktiviert:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE CONFIGURATION>
     <CONFIGURATION>
       <OnlineHelpFiles>
         <hmi_myhelp>
           <EntriesFile value="hmi_myhelp.xml" type="QString"/>
           <DisableIndex value="true" type="bool"/>
         </hmi_myhelp>
       </OnlineHelpFiles>
     </CONFIGURATION>
```

3.7.2 Aufbau und Syntax des Hilfebuchs

Syntax für das Hilfebuch

Das Hilfebuch ist eine XML-Datei, in der der Aufbau der Online-Hilfe festgelegt ist. Der Name der Datei ist frei wählbar, z. B. "hmi_myhelp". In dieser Datei definieren Sie:

- HTML-Dokumente
- Inhalts- und Stichwortverzeichnis

Tag	Anzahl	Bedeutung
HMI_SL_HELP	1	Root Element des XML-Dokuments
BOOK		Bezeichnet ein Hilfebuch. Der Name ist frei wählbar unter der Randbedingung, dass kein vom System vordefinierter Name (wie z. B. sinumerik_alarm_plc_pmc) verwendet werden darf. Attribute:

Tag	Anzahl	Bedeutung	
	+	ref	Bezeichnet das HTML-Dokument, das als Einstiegsseite für das Hilfebuch angezeigt wird.
		titel	Titel des Hilfebuchs, das im Inhaltsverzeichnis angezeigt wird.
		helpdir	Verzeichnis, das die Online-Hilfe des Hilfebuchs beinhaltet.
ENTRY	*	Kapitel der Online-Hilfe Attribute:	
		ref	Bezeichnet das HTML-Dokument, das als Einstiegsseite für das Kapitel angezeigt wird.
		titel	Titel des Kapitels, das im Inhaltsverzeichnis angezeigt wird.
INDEX_ENTRY	*	Anzuzeigendes Stichwort Attribute:	
		ref	Bezeichnet das HTML-Dokument, das für diesen Stichworteintrag angesprungen wird.
		titel	Titel des Stichworts, das im Stichwortverzeichnis angezeigt wird.

Für die Spalte "Anzahl" gilt:

* bedeutet 0 oder mehrere

+ bedeutet 1 oder mehrere

Index formatieren

Sie haben folgende Möglichkeiten, das Stichwortverzeichnis zu formatieren:

- Einzeleintrag: <INDEX_ENTRY ...title="index"/>
- Zwei zweistufige Einträge, wobei jeder Titel einen Haupt- und Untereintrag hat.

  Trennen Sie die Einträge mit einem Komma voneinander.

  ```
  <INDEX_ENTRY ...title="mainIndex_1,subIndex_1 with mainIndex_1"/>
  ```
- Zweistufiger Eintrag, wobei der erste Titel der Haupteintrag und der zweite Titel der Untereintrag ist.

  Trennen Sie die Einträge mit einem Semikolon voneinander.

  ```
  <INDEX_ENTRY ...title="mainIndex_2;subIndex_2

  without mainIndex_1"/>
  ```

- index
- mainIndex_1
  - subIndex_1 with mainIndex_1
- mainIndex_2
  - subIndex_2 without mainIndex_2
- subIndex_1 with mainIndex_1
  - mainIndex_1

Bild 3-9 Beispiel: zweistufiger Index

3.7.3 Beschreibung der Syntax für die Online-Hilfe

Regeln für die Erstellung der HTML-Dateien

Erzeugen Sie die Hilfedateien im HTML-Format. Es ist dabei möglich, alle Informationen in einer einzelnen HTML-Datei abzulegen oder auf mehrere HTML-Dateien zu verteilen.

Die Dateinamen sind unter Beachtung der folgenden Regeln zu vergeben:

- Verweise innerhalb der HTML-Dateien sollen immer mit relativen Pfaden angeben. Nur so ist sichergestellt, dass die Verweise sowohl auf dem Entwicklungsrechner, als auch auf dem Zielsystem gleichermaßen funktionieren.
- Wenn zu bestimmten Punkten innerhalb einer HTML-Datei per Link gesprungen werden soll, müssen dafür sogenannte Anker definiert werden.

  Beispiel für einen HTML-Anker:

  ```
  <a name="myAnchor">This is an anchor</a>
  ```

  Siehe auch: Beispiel: So erstellen Sie eine Online-Hilfe für Anwender PLC-Alarme (Seite 75)
- Der Inhalt der HTML-Dokumente muss mit der Codierung UTF-8 abgelegt werden. Dadurch ist gewährleistet, dass die HTML-Dokumente in allen unterstützten Landessprachen korrekt angezeigt werden.

HTML-Tags

Es wird folgende Untermenge des HTML-Funktionsumfangs unterstützt:

Tag	Beschreibung	Kommentar
a	Anchor or link	Unterstützte Attribute: href und name
address	Address	
b	Bold	
big	Larger font	
blockquote	Indented paragraph	
body	Document body	Unterstützte Attribute: bgcolor (#RRGGBB)
br	Line break	
center	Centered paragraph	
cite	Inline citation	Gleiche Wirkung wie Tag i
code	Code	Gleiche Wirkung wie Tag tt
dd	Definition data	
dfn	Definition	Gleiche Wirkung wie Tag i
div	Document division	Unterstützt werden die Standard-Satzattribute
dl	Definition list	Unterstützt werden die Standard-Satzattribute
dt	Definition term	Unterstützt werden die Standard-Satzattribute
em	Emphasized	Gleiche Wirkung wie Tag i
font	Font size, family, color	Unterstützte Attribute: size, face, and color (#RRGGBB)
h1	Level 1 heading	Unterstützt werden die Standard-Satzattribute

Tag	Beschreibung	Kommentar
h2	Level 2 heading	Unterstützt werden die Standard-Satzattribute
h3	Level 3 heading	Unterstützt werden die Standard-Satzattribute
h4	Level 4 heading	Unterstützt werden die Standard-Satzattribute
h5	Level 5 heading	Unterstützt werden die Standard-Satzattribute
h6	Level 6 heading	Unterstützt werden die Standard-Satzattribute
head	Document header	
hr	Horizontal line	Unterstützte Attribute: width (kann als absoluter oder relativer Wert angegeben werden)
html	HTML document	
i	Italic	
img	Image	Unterstützte Attribute: src, width, height
kbd	User-entered text	
meta	Meta-information	
li	List item	
nobr	Non-breakable text	
ol	Ordered list	Unterstützt werden die die Standardattribute für Listen
p	Paragraph	Unterstützt werden die Standard-Satzattribute (Voreinstellung: left-aligned)
pre	Preformated text	
s	Strikethrough	
samp	Sample code	Gleiche Wirkung wie Tag tt
small	Small font	
span	Grouped elements	
strong	Strong	Gleiche Wirkung wie Tag b
sub	Subscript	
sup	Superscript	
table	Table	Unterstützte Attribute: border, bgcolor (#RRGGBB), cellspacing, cellpadding, width (absolut oder relative), height
tbody	Table body	Wirkungslos
td	Table data cell	Unterstützt werden die die Standardattribute für Tabellenzellen
tfoot	Table footer	Wirkungslos
th	Table header cell	Unterstützt werden die die Standardattribute für Tabellenzellen
thead	Table header	Wird für das Drucken von Tabellen verwendet, die sich über mehrere Seiten erstrecken
title	Document title	
tr	Table row	Unterstützte Attribute: bgcolor (#RRGGBB)
tt	Typewrite font	
u	Underlined	
ul	Unordered list	Unterstützt werden die die Standardattribute für Listen
var	Variable	Gleiche Wirkung wie Tag tt

Satzattribute

Folgende Attribute werden von den Tags div, dl, dt, h1, h2, h3, h4, h5, h6, p unterstützt:

- align (left, right, center, justify)
- dir (ltr, rtl)

Standardattribute für Listen

Folgende Attribute werden von den Tags ol und ul unterstützt:

- type (1, a, A, square, disc, circle)

Standardattribute für Tabellen

Folgende Attribute werden von den Tags td und th unterstützt:

- width (absolute, relative, no-value)
- bgcolor (#RRGGBB)
- colspan
- rowspan
- align (left, right, center, justify)
- valign (top, middle, bottom)

CSS-Eigenschaften

Die nachfolgende Tabelle enthält den unterstützten CSS-Funktionsumfang:

Property	Werte	Beschreibung
background-color	<color>	Hintergrundfarbe für Elemente
background-image	<uri>	Hintergrundbild für Elemente
color	<color>	Vordergrundfarbe für Text
text-indent	<length>px	Einrückung der ersten Zeile eines Absatzes in Pixel
white-space	normal | pre | nowrap | pre-wrap	Bestimmt wie Whitespace-Zeichen in HTML-Dokumenten behandelt werden
margin-top	<length>px	Breite des oberen Absatzrandes in Pixel
margin-bottom	<length>px	Breite des unteren Absatzrandes in Pixel
margin-left	<length>px	Breite des linken Absatzrandes in Pixel
margin-right	<length>px	Breite des rechten Absatzrandes in Pixel
vertical-align	baseline | sub | super | middle | top | bottom	Vertikale Ausrichtung für Text (in Tabellen werden nur die Werte middle, top, und bottom unterstützt)
border-color	<color>	Randfarbe für Texttabellen

Property	Werte	Beschreibung
border-style	none | dotted | dashed | dot-dash | dot-dot-dash | solid | double | groove | ridge | inset | outset	Randstil für Texttabellen
background	[<'background-color'> || <'background-image'>]	Kurzschreibweise für background Property
page-break-before	[auto | always]	Seitenumbruch vor einem Absatz/einer Tabelle
page-break-after	[auto | always]	Seitenumbruch nach einem Absatz/einer Tabelle
background-image	<uri>	Hintergrundbild für Elemente

Unterstützte CSS-Selektoren

Alle CSS 2.1 Selektorklassen werden unterstützt mit Ausnahme von sog. Pseudo-Selektorklassen wie :first-child, :visited und :hover.

3.7.4 Beispiel: So erstellen Sie ein OEM-spezifisches Online-Hilfebuch

Voraussetzungen

Erstellen Sie folgende Dateien:

- Konfigurationsdatei: "slhlp.xml"

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE CONFIGURATION>
     <CONFIGURATION>
       <OnlineHelpFiles>
         <hmi_myhelp>
           <EntriesFile value="hmi_myhelp.xml" type="QString"/>
           <DisableIndex value="false" type="bool"/>
         </hmi_myhelp>
       </OnlineHelpFiles>
     </CONFIGURATION>
```

- Definition des Hilfebuchs: "hmi_myhelp.xml"

```
<?xml version="1.0" encoding="UTF-8"?>
<HMI_SL_HELP language="en-US">
     <BOOK ref="index.html" title="Easy Help" helpdir="hmi_myhelp">
       <ENTRY ref="chapter_1.html" title="Chapter 1">
```

```
          <INDEX_ENTRY ref="chapter_1.html" title="Keyword 1"/>
          <INDEX_ENTRY ref="chapter_1.html" title="Keyword 2"/>
        </ENTRY>
        <ENTRY ref="chapter_2.html" title="Chapter 2">
          <INDEX_ENTRY ref="chapter_2.html" title="Keyword 2"/>
        </ENTRY>
        <ENTRY ref="chapter_3.html" title="Chapter 3">
          <INDEX_ENTRY ref="chapter_3.html" title="Keyword 3"/>
          <ENTRY ref="chapter_31.html" title="Chapter 3.1">
            <INDEX_ENTRY ref="chapter_31.html" title="test;Chapter 3.1"/>
          </ENTRY>
          <ENTRY ref="chapter_32.html" title="Chapter 3.2">
            <INDEX_ENTRY ref="chapter_32.html" title="test;Chapter 3.2"/>
          </ENTRY>
        </ENTRY>
      </BOOK>
</HMI_SL_HELP>
```

Hilfedateien im Zielsystem ablegen

Im nachfolgenden Beispiel wird der Aufbau eines Hilfebuchs mit Namen "Easy Help" mit Inhaltsverzeichnis und Stichwortverzeichnis beschrieben.

Vorgehensweise:

1. Kopieren Sie die Konfigurationsdatei "slhlp.xml" in folgendes Verzeichnis:

   `/oem/sinumerik/hmi/cfg`

2. Legen Sie für die gewünschte Sprache der Online-Hilfe ein Verzeichnis unter folgendem Pfad an: `/oem/sinumerik/him/hlp`

   Verwenden Sie dafür die vorgegebene Sprachkennung aus Kapitel Liste der Sprachkennzeichen für Dateinamen (Seite 403).

   Hinweis

   Schreibweise

   Die Namen der Verzeichnisse müssen unbedingt kleingeschrieben werden.

   Wenn Sie z. B. eine Hilfe in Englisch einbinden, legen Sie einen Ordner "eng" an.

   ---

3. Legen Sie das Hilfebuch z. B. "hmi_myhelp.xml" in den Ordner "eng":

   `/oem/sinumerik/him/hlp/eng/hmi_myhelp.xml`

4. Kopieren Sie die Hilfedateien in folgendes Verzeichnis:

   `/oem/sinumerik/him/hlp/eng/hmi_myhelp/`

Die Einstellungen werden erst nach dem Neustart des Systems wirksam.

Hinweis

Aktualisierungen oder Änderungen

Bei der Anzeige des Inhalts- und Stichwortverzeichnisses eines Hilfebuchs werden zur schnelleren Bearbeitung im Verzeichnis `/siemens/sinumerik/sys_cache/hmi/hlp` die Hilfedateien im Binärformat abgelegt: slhlp_<Hilfebuch>_*_<lng>.hmi .

Im Beispiel: slhlp_hmi_myhelp_*_eng.hmi

Damit die Änderungen in der Online-Hilfe wirksam und angezeigt werden, sind diese Dateien vorher zu löschen.

Ergebnis

Das Buch besteht aus drei Kapiteln mit Unterkapiteln:

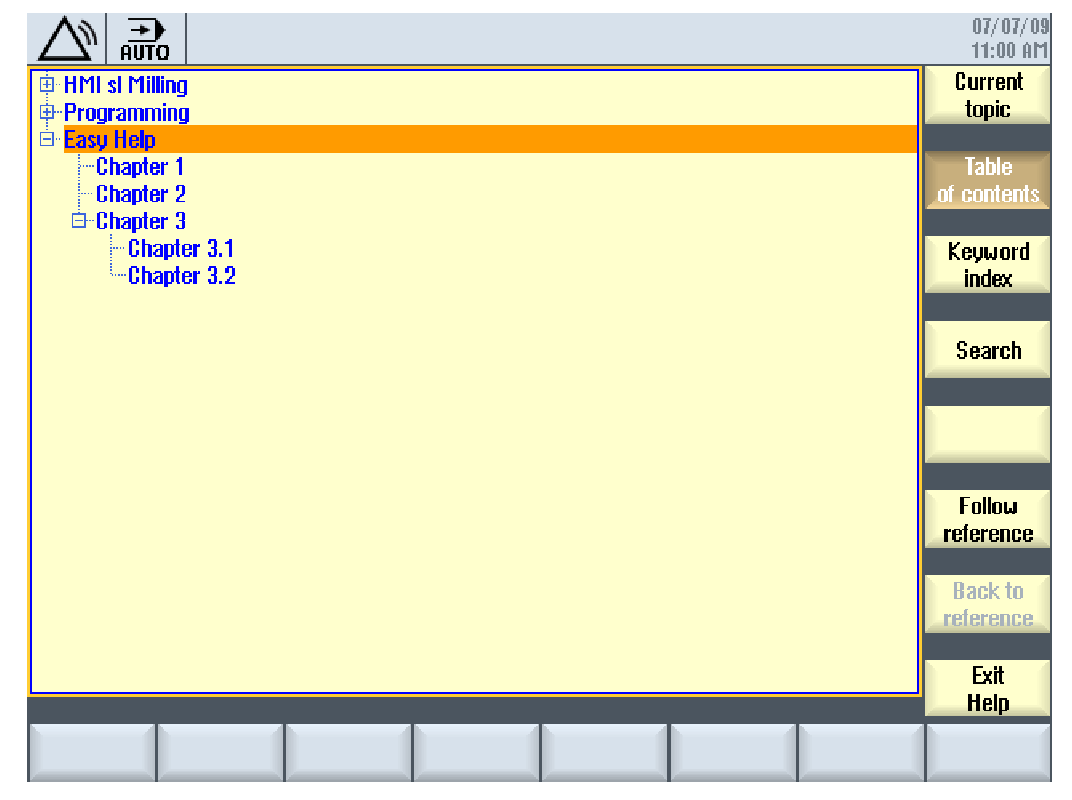


Bild 3-10 Beispiel: OEM Online-Hilfe

Einträge im Stichwortverzeichnis:

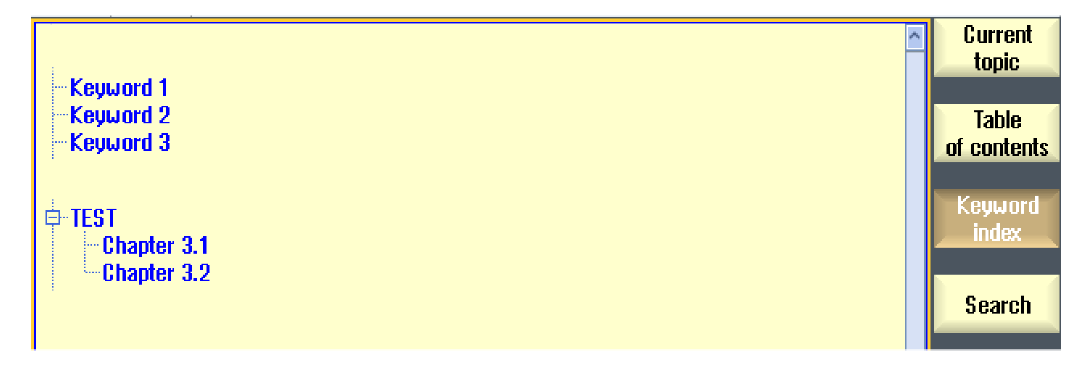


Bild 3-11 Beispiel: Index

3.7.5 Beispiel: So erstellen Sie eine Online-Hilfe für Anwender PLC-Alarme

Übersicht

Wird ein Anwender PLC-Alarm ausgelöst, kann eine kontextsensitive Online-Hilfe zum jeweiligen Alarm z. B. mit Erläuterung und Abhilfe zur Verfügung erstellt werden. Die Online-Hilfetexte für die Anwender PLC-Alarme werden in folgender Datei verwaltet: "sinumerik_alarm_oem_plc_pmc.html"

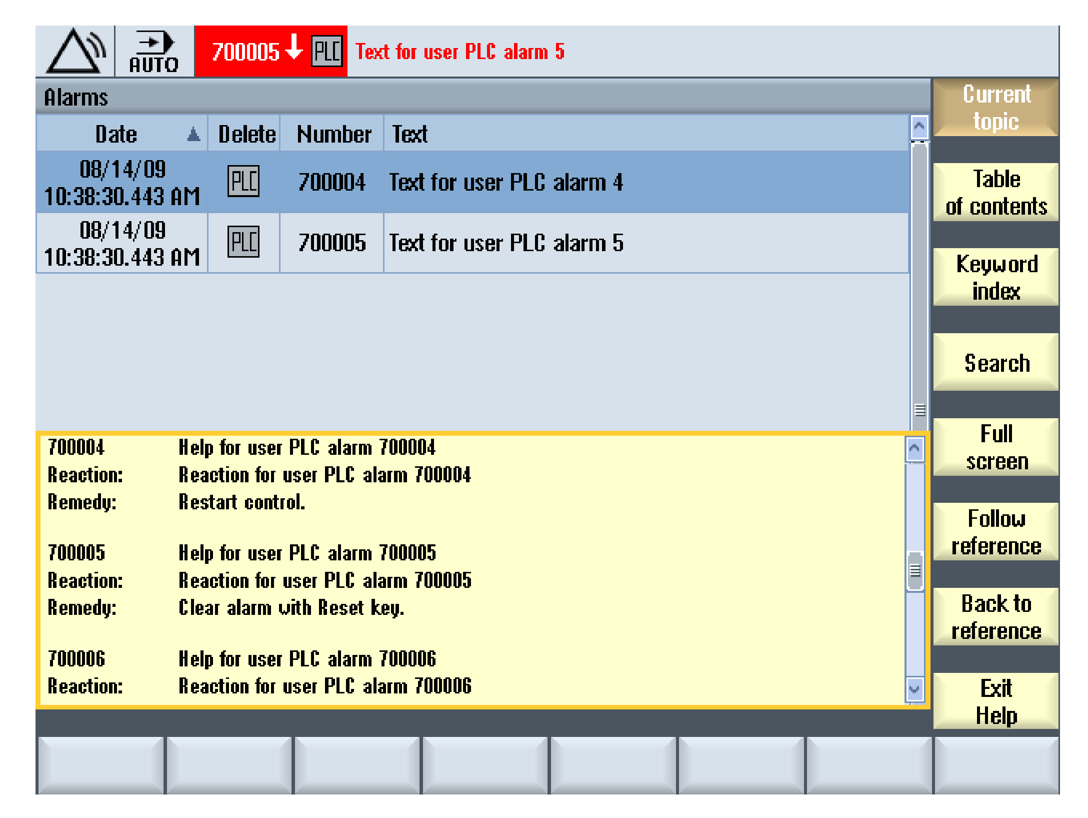


Bild 3-12 Beispiel: Online-Hilfe für Anwender PLC-Alarme

Aufbau der Hilfedatei

Folgende Einträge sind in der Hilfedatei zulässig:

Eintrag	Bedeutung
AlarmNr	Hyperlink zur Alarmnummer
	Hilfetext für den entsprechenden Alarm
<td width="85%">......</td>	Text der hinter dem Feld "Erläuterung" oder "Abhilfe" angezeigt wird.

Hilfedatei erstellen

Der Dateiname ist **sprachunabhängig** und muss heißen:

sinumerik_alarm_oem_plc_pmc.html

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE html PUBLIC>

<html>
<head><title></title></head>
<body>
<table>
...
<tr>
<td width="15%">
<b><a name="700004">700004</a></b></td>
<td width="85%"><b>Help for user PLC alarm 700004 </b></td></tr>
<tr><td valign="top" width="15%"><b>Reaction: </b></td>
<td width="85%">Reaction for user PLC alarm 700004 </td></tr>
<tr><td valign="top" width="15%"><b>Remedy:</b></td>
<td width="85%">Restart control. </td>
</tr>
<br>

<tr>
<td width="15%">
<b><a name="700005">700005</a></b></td>
<td width="85%"><b>Help for user PLC alarm 700005 </b></td></tr>
<tr><td valign="top" width="15%"><b>Reaction: </b></td>
<td width="85%">Reaction for user PLC alarm 700005 </td></tr>
<tr><td valign="top" width="15%"><b>Remedy:</b></td>
<td width="85%">Clear alarm with Reset key. </td>
</tr>
<br>
```

sinumerik_alarm_oem_plc_pmc.html

```
...
</table>
<p></p>
</body>
</html>
```

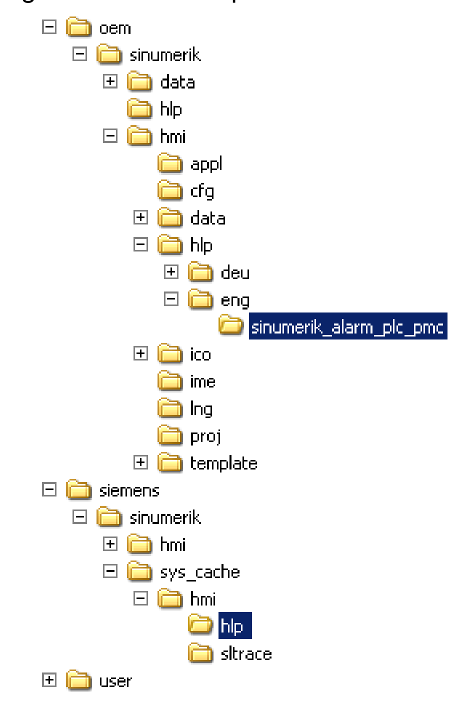
Vorgehensweise:

1. Kopieren Sie die Datei in eines der folgenden Verzeichnisse:

   `/oem/sinumerik/hmi/hlp/<lng>/sinumerik_alarm_plc_pmc/`

   `/user/sinumerik/hmi/hlp/<lng>/sinumerik_alarm_plc_pmc/`

   <lng> steht für das Sprachkennzeichen.

2. Löschen Sie alle Dateien im Verzeichnis:

   `/siemens/sinumerik/sys_cache/hmi//hlp`

Die Einstellungen werden erst nach dem Neustart des Systems wirksam.

3.7.6 Beispiel: So erstellen Sie eine Online-Hilfe für NC-/PLC-Variablen

Übersicht

Um eine kontextsensitive Online-Hilfe zu NC-/PLC-Variablen oder Systemvariablen wie im nachfolgenden Beispiel zu erstellen, werden die Beschreibungstexte in sprachabhängigen html-Dateien verwaltet.

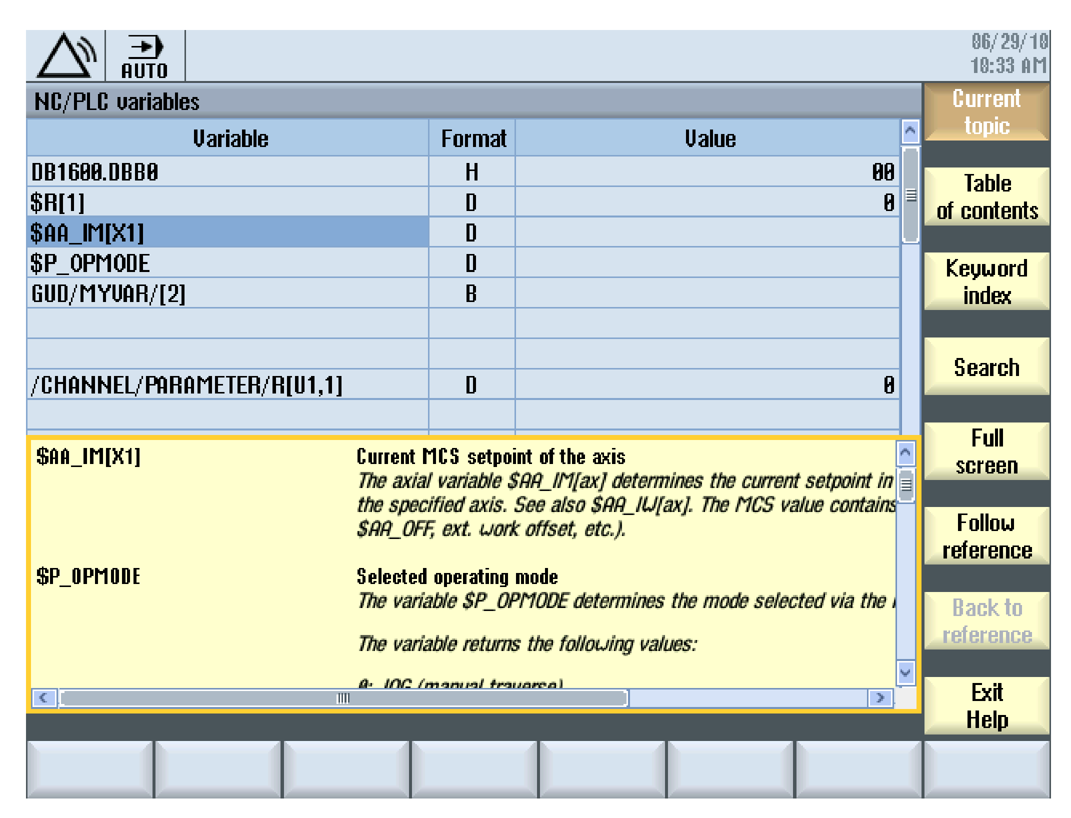


Bild 3-13 Beispiel: Online-Hilfe für Anwender-Variablen

Aufbau der Online-Hilfe

Für die Online-Hilfe werden folgende Dateien benötigt:

Datei	Bedeutung
`sldgvarviewhelp.ini`	Konfigurationsdatei zur Verwaltung der html-Datei oder mehrerer html-Dateien
`<lng>/<name>1.html`	Die Inhalte der html-Dateien der Online-Hilfe sind sprachabhängig und werden im jeweiligen Sprachverzeichnis <lng> abgelegt.
`<lng>/<name>2.html`	
. . .	
`<lng>/<name>n.html`	

Aufbau der Konfigurationsdatei

Die Datei ist **sprachunabhängig** und muss heißen:

sldgvarviewhelp.ini

```
[HelpBindings]
/BAG/STATE/OPMODE = var1_help.html#var1
$AA_IM[X1] = var1_help.html
$R[1] = var1_help.html#var2
/Channel/Parameter/R[u1,1] = var2_help.html#var2
DB1600.DBB0 = var2_help.html#var1
GUD/MyVar[2] = var2_help.html
```

Hinweis

Die html-Dateien können mit einem beliebigen html-Editor erstellt werden. In der Konfigurationsdatei wird definiert, welche html-Dateien zur Online-Hilfe gehören.

Die Beschreibung kann aus einer oder mehreren html-Dateien bestehen: zum Beispiel pro Variable eine html-Datei oder mehrere gleichartige Variablen in eine Datei.

Vorgehensweise:

1. Kopieren Sie die Konfigurationsdatei in das folgende Verzeichnis:

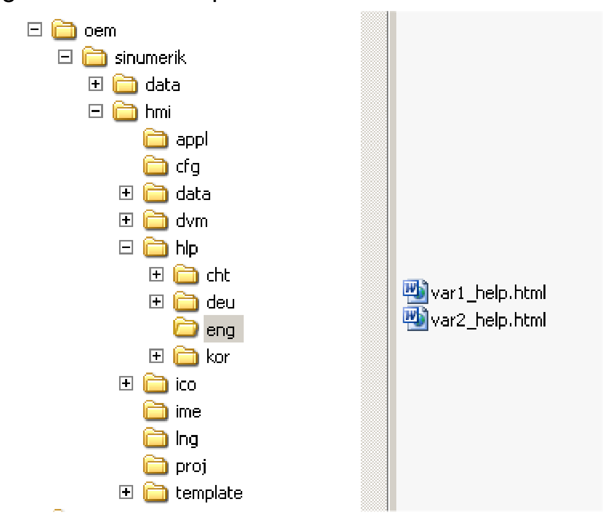
```
/oem/sinumerik/hmi/cfg/sldgvarviewhelp.ini
```

2. Kopieren Sie die html-Dateien in eines der folgenden Verzeichnisse:

```
/oem/sinumerik/hmi/hlp/<lng>/
```

```
/user/sinumerik/hmi/hlp/<lng>/
```

<lng> steht für das Sprachkennzeichen.



3. Löschen Sie alle Dateien im Verzeichnis:

   `/siemens/sinumerik/sys_cache/hmi//hlp`

Die Einstellungen werden erst nach dem Neustart des Systems wirksam.

3.7.7 Beispiel: So erstellen Sie eine Programmier-Online-Hilfe

Voraussetzungen

Erstellen Sie folgende Dateien:

- Konfigurationsdatei "prog_help.ini"

```
[milling]
CYCLE1=cycle1_help.html
CYCLE2=cycle2_help.html#TextAnchor1
CYCLE3=cycle3_help.html
CYCLE4=cycle4_help.html

[turning]
CYCLE3=cycle2_help.html
CYCLE4=cycle3_help.html
```

- Konfigurationsdatei für das Hilfebuch "slhlp.xml" (optional)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE CONFIGURATION>
<CONFIGURATION>
  <OnlineHelpFiles>
    <hmi_prog_help>
      <EntriesFile value="hmi_prog_help.xml" type="QString"/>
      <DisableIndex value="true" type="bool"/>
    </hmi_prog_help>
  </OnlineHelpFiles>
</CONFIGURATION>
```

- Konfigurationsdatei für das Hilfebuch "hmi_prog_help.xml" (optional)

```
<?xml version="1.0" encoding="UTF-8"?>
<HMI_SL_HELP language="en-US">
  <BOOK ref="index.html" title="OEM_CYCLES" helpdir="hmi_prog_help">
    <ENTRY ref="cycle1_help.html" title="Cycle1"></ENTRY>
    <ENTRY ref="cycle2_help.html" title="Cycle2"></ENTRY>
    <ENTRY ref="cycle3_help.html" title="Cycle3"></ENTRY>
    <ENTRY ref="cycle4_help.html" title="Cycle4"></ENTRY>
    <ENTRY ref="cycle_help.html" title="OEM_Cycles"></ENTRY>
  </BOOK>
</HMI_SL_HELP>
```

- Sprachabhängige Datei "<prog_help_eng>.ts": dieser Dateiname ist fest vorgegeben.

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<!DOCTYPE TS>
<TS>
     <context>
       <name>oem_cycles</name>
         <message>
           <source>CYCLE1</source>
           <translation>short description for cycle 1</translation>
           <chars>*</chars>
         </message>
         <message>
           <source>CYCLE2</source>
           <translation>short description for cycle 2</translation>
           <chars>*</chars>
         </message>
         <message>
           <source>CYCLE3</source>
           <translation>short description for cycle 3</translation>
           <chars>*</chars>
         </message>
     </context>
</TS>
```

Hilfedateien im Zielsystem ablegen

Vorgehensweise:

1. Kopieren Sie die Konfigurationsdatei "prog.ini" in folgendes Verzeichnis:

   `/oem/sinumerik/hmi/cfg`

2. Kopieren Sie die Datei "`slhlp.xml`" für das Hilfebuch in folgendes Verzeichnis:

   `/oem/sinumerik/hmi/cfg`

3. Legen Sie für die gewünschte Sprache der Online-Hilfe ein Verzeichnis unter folgendem Pfad an: `/oem/sinumerik/hmi/hlp/<lng>` und kopieren sie dorthin die Datei `hmi_prog_help.xml`

   Verwenden Sie dafür die vorgegebene Sprachkennung aus Liste der Sprachkennzeichen für Dateinamen. Die Namen der Verzeichnisse müssen unbedingt kleingeschrieben werden.

4. Die sprachabhängige Datei `prog_help_<lng>.ts` für die Kurzbeschreibung kopieren Sie in folgendem Pfad: `/oem/sinumerik/hmi/<lng>/prog_help_<lng>.ts`

5. Kopieren Sie die html-Dateien mit der Beschreibung der OEM-Zyklen in folgendes Verzeichnis:

   `/oem/sinumerik/him/hlp/<lng>/hmi_prog_help/cycle<n>_help.html`

Die Einstellungen werden erst nach dem Neustart des Systems wirksam.

Siehe auch

Beispiel: So erstellen Sie ein OEM-spezifisches Online-Hilfebuch (Seite 72)

Peripherie in Betrieb nehmen 4

4.1 Anschließen der Peripheriemodule

4.1.1 Peripheriemodule aktivieren

Maschinendaten für die PLC-Peripherie

Folgende Komponenten belegen feste Adressen für das Eingangs- und Ausgangsabbild der PLC: Peripheriemodule, Maschinensteuertafel, SENTRON PAC3200, SENTRON PAC4200 und PN-Buskoppler.

Die Maschinendaten enthalten zwei Felder, um die Aktualisierung des Eingangs- und Ausgangsabbilds der PLC zu deaktivieren:

Maschinendatum		Wertebereich	
12986[i]	$MN_PLC_DEACT_IMAGE_LADDR_IN	0 ≤ i ≤ 7	Eingangsadressen
12987[i]	$MN_PLC_DEACT_IMAGE_LADDR_OUT	0 ≤ i ≤ 7	Ausgangsadressen

Die SINUMERIK 828D arbeitet mit einer festen Maximalkonfiguration der Peripheriemodule. Im Auslieferungszustand ist die Datenübertragung zum Eingangs- und Ausgangsabbild der PLC für alle Peripheriemodule deaktiviert.

Feld mit logischen Eingangsadressen:

MD	Logische Eingangsadresse	Datenübertragung zur PLC deaktiviert
12986[0]	0	1. PP-Modul inaktiv
12986[1]	9	2. PP-Modul inaktiv
12986[2]	18	3. PP-Modul inaktiv
12986[3]	27	4. PP-Modul inaktiv
12986[4]	36	5. PP-Modul inaktiv
12986[5]	96	PN-Buskoppler inaktiv
12986[6]	112	Maschinensteuertafel inaktiv
12986[7]	132	SENTRON PAC4200 inaktiv
12986[8]	144	SENTRON PAC3200 inaktiv

Das Feld der Ausgangsadressen ist leer (Voreinstellung): MD12987[i] = -1

Soll ein Peripheriemodul aktiviert werden, darf seine Adresse weder in MD12986[i] noch in MD12987[i] eingetragen sein. Stattdessen ist der Wert -1 ("leer") einzutragen.

Beispiel

Es sind zwei PP-Module, die Maschinensteuertafel und das SENTRON PAC4200 aktiviert:

MD	Logische Eingangsadresse	Datenübertragung zur PLC deaktiviert
12986[0]	**-1**	**1. PP-Modul aktiv**
12986[1]	**-1**	**2. PP-Modul aktiv**
12986[2]	18	3. PP-Modul inaktiv
12986[3]	27	4. PP-Modul inaktiv
12986[4]	36	5. PP-Modul inaktiv
12986[5]	96	PN-Buskoppler inaktiv
12986[6]	**-1**	**Maschinensteuertafel aktiv**
12986[7]	**-1**	**SENTRON PAC4200 aktiv**
12986[8]	144	SENTRON PAC3200

Hinweis

Die Verwendung einer Ein-/Ausgangsadresse eines deaktivierten Moduls im PLC-Anwenderprogramm löst keinen Alarm aus. Das PLC-Anwenderprogramm arbeitet immer mit dem Abbildspeicher. Ob es eine Verbindung zu physikalischen Ein-/Ausgängen gibt, wird über MD12986[i] und MD12987[i] konfiguriert.

Aktive Module werden dann zyklisch auf Ausfall überwacht.

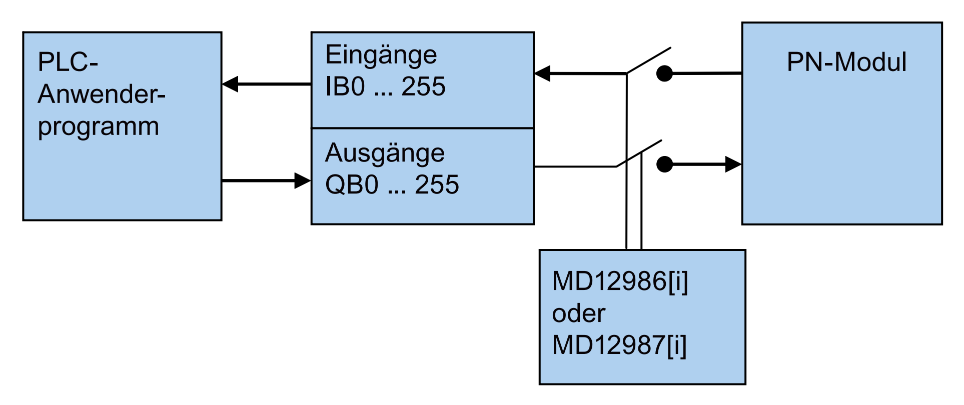


Bild 4-1 Peripherie-Schalter

4.1.2 Adressierung der Peripheriemodule

IP-Adressen der Peripheriemodule

Nachfolgender Tabelle ist zu entnehmen, welcher Einstellung des DIP-Schalters S1 die zugehörige IP-Adresse des jeweiligen Peripheriemoduls entspricht. Dabei ist der Maximalausbau mit Peripheriemodulen, Buskoppler und Maschinensteuertafel über die PLC E/A-Schnittstelle auf Basis PROFINET berücksichtigt.

Peripheriemodul	Bus	Gerätename	IP-Adresse	Input-Adressen	Output-Adressen
			192.168.214.	(aktiv mit MD12986[x] = -1)	
				Index n:	
1. PP-Modul digital	PN	pp72x48pn9	9	0 … 8	0 … 5
2. PP-Modul digital	PN	pp72x48pn8	8	9 … 17	6 … 11
3. PP-Modul digital	PN	pp72x48pn7	7	18 … 26	12 … 17
4. PP-Modul digital	PN	pp72x48pn6	6	27… 35	18 … 23
5. PP-Modul digital	PN	pp72x48pn5	5	36 … 44	24 … 29
frei				45	30 … 55
				Index d:	
1. PP-Modul Diagnose	PN	pp72x48pn9	9	46 … 47	--
2. PP-Modul Diagnose	PN	pp72x48pn8	8	48 … 49	--
3. PP-Modul Diagnose	PN	pp72x48pn7	7	50 … 51	--
4. PP-Modul Diagnose	PN	pp72x48pn6	6	52 … 53	--
5. PP-Modul Diagnose	PN	pp72x48pn5	5	54 … 55	--
				Index m:	
1. PP-Modul analog	PN	pp72x48pn9	9	56 … 63	56 … 63
2. PP-Modul analog	PN	pp72x48pn8	8	64 … 71	64 … 71
3. PP-Modul analog	PN	pp72x48pn7	7	72 … 79	72 … 79
4. PP-Modul analog	PN	pp72x48pn6	6	80 ... 87	80 ... 87
5. PP-Modul analog	PN	pp72x48pn5	5	88 … 95	88 … 95
PN/PN-Koppler *	PN	pn-pn-coupler20	20	96 … 111	96 … 111
externe Maschinensteuertafel	PN	mcp-pn64	64	112 … 125	112 … 121
reserviert		--	--	126 … 131	122 ... 123
Sentron PAC 4200 *	PN	pac4200-pn21	21	132 ... 143	132 ... 143
Sentron PAC 3200 *	PN	pac3200-pn22	22	144 ... 155	144 ... 155

Der Index n, m, d ist immer die Anfangsadresse des Adressbereiches.

*) Die IP-Adresse dieser Komponenten wird nicht über Schalterstellung, sondern bei der Projektierung eingestellt.

5 Maschinendaten parametrieren

5.1 Einteilung der Maschinendaten

Berechtigung für Maschinendaten

Um Maschinendaten einzugeben oder zu ändern, ist mindestens das Kennwort für Hersteller erforderlich.



Gefahr für Leben und Maschine

Veränderungen von Maschinendaten haben einen wesentlichen Einfluss auf die Maschine.

Fehlerhafte Parametrierung kann Menschenleben gefährden und zur Zerstörung der Maschine führen.

Einteilung der Maschinendaten

Die Maschinendaten sind in folgende Bereiche eingeteilt:

- Allgemeine Maschinendaten ($MN)
- Kanalspezifische Maschinendaten ($MC)
- Achsspezifische Maschinendaten ($MA)
- Anzeige-Maschinendaten ($MM)
- Allgemeine Setting-Daten ($SN)
- Kanalspezifische Setting-Daten ($SC)
- Achsspezifische Setting-Daten ($SA)
- SINAMICS Maschinendaten (Control Unit und Antriebs-Maschinendaten):

  r0001 ... r9999 (nur lesen)

  p0001 ... p9999 (lesen und schreiben)

Hinweis

Die Maschinendaten für die Technologien Drehen und Fräsen sind bereits so voreingestellt, dass eine Anpassung der Maschinendaten nur in Ausnahmefällen erforderlich ist.

Für jeden dieser Bereiche existiert ein eigenes Listenabbild, in dem Sie die Maschinendaten ansehen und ändern können:



Bild 5-1 Softkey-Leiste

Folgende Eigenschaften der Maschinendaten werden von links nach rechts angezeigt:

- Nummer des Maschinendatums ggf. mit Feldindex in eckigen Klammern
- Name des Maschinendatums
- Wert des Maschinendatums
- Einheit des Maschinendatums
- Wirksamkeit des Maschinendatums
- Datenklasse

Physikalische Einheiten der Maschinendaten

Die physikalischen Einheiten von Maschinendaten werden rechts neben dem Eingabefeld angezeigt:

Anzeige	Einheit	Meßgröße
m/s**2	m/s²	Beschleunigung
U/s**3	U/s³	Änderung der Beschleunigung für die drehende Achse
kg/m**2	kgm²	Trägheitsmoment
mH	mH (Milli-Henry):	Induktivität
Nm	Nm (Newton-Meter):	Drehmoment
us	µs (Mikro-Sekunde):	Zeit
uA	µA (Mikro-Ampere):	Stromstärke
uVs	µVs (Mikro-Volt-Sekunde):	Magnetischer Fluss
userdef	anwenderdefiniert:	Die Einheit wird vom Anwender festgelegt.

Bei Maschinendaten ohne Einheit ist die Einheitenspalte leer.

Sind die Daten nicht verfügbar, wird anstelle des Wertes "#" angezeigt. Ist der Wert mit einem "H" abgeschlossen, handelt es sich um Werte in Hexadezimal-Darstellung.

Wirksamkeit der Maschinendaten

In der rechten Spalte wird angezeigt, wann die Änderung des Maschinendatums wirksam wird:

cf	Die geänderten Maschinendaten erfordern eine Aktivierung durch den Softkey "MD wirksam setzen".
po	Die geänderten Maschinendaten erfordern einen "NCK Power On-Reset", um wirksam zu werden.
re	Die geänderten Maschinendaten erfordern eine Aktivierung durch die Taste <RESET>.
so	Die geänderten Maschinendaten sind sofort aktiv.

Anwendersichten

Anwendersichten sind anwenderspezifische Sammlungen von Maschinendaten. Sie dienen dazu, alle in einem bestimmten Bedienzustand relevanten Maschinendaten aus verschiedenen Bereichen zur Bearbeitung zur Verfügung zu stellen.

Die Anwendersichten werden unter folgendem Pfad auf der CompactFlash Card gespeichert:

`user/sinumerik/hmi/template/user_views`

Als Vorlage sind bereits folgende Anwendersichten vorhanden:

- Electrical_Startup
- Mechanical_Startup
- Optimizing_Axis

Siehe auch

Beschreibung der Datenklassen: Funktionshandbuch Grundmaschine (P4)

Eine ausführliche Beschreibung der Maschinendaten finden Sie im Listenhandbuch mit Querverweisen auf das entsprechende Kapitel des Funktionshandbuchs.

5.2 Teileprogramme von externen CNC-Systemen verarbeiten

Hinweis

Voreinstellung

Bei SINUMERIK 828D sind die folgenden Maschinendaten bereits für die jeweilige Technologie Drehen oder Fräsen geeignet voreingestellt. Diese Maschinendaten sind nicht änderbar und werden nicht angezeigt.

Funktion ISO Dialekt aktivieren

Mit dem Maschinendatum MD18800 $MN_EXTERN_LANGUAGE wird die externe Sprache aktiviert. Die Auswahl des Sprachtyps ISO-Dialekt M oder T erfolgt über das MD10880 $MN_EXTERN_CNC_SYSTEM.

Die Umschaltung von Siemens-Modus nach ISO-Dialekt-Modus erfolgt durch die beiden G-Befehle der Gruppe 47:

- G290: NC-Programmiersprache Siemens aktiv
- G291: NC-Programmiersprache ISO-Dialekt aktiv

Dabei bleiben das aktive Werkzeug, die Werkzeugkorrekturen und Nullpunktverschiebungen erhalten. G290 und G291 müssen allein in einem NC-Programmsatz stehen.

Die Umschaltung auf eine externe Programmiersprache ist bei SINUMERIK 828D im Lieferumfang enthalten. MD10712 $MN_NC_USER_CODE_CONF_NAME_TAB gilt nur für NC-Sprachbefehle im Siemens-Modus.

Literatur

Funktionshandbuch ISO Dialekte

5.3 Freiformflächen mit Advanced Surface bearbeiten

Bearbeitung mit Advanced Surface

Beim Abarbeiten von CAM-Programmen im Bereich von High Speed Cutting (HSC) müssen von der Steuerung hohe Vorschübe bei kürzesten NC-Sätzen verarbeitet werden. Dabei soll im Ergebnis das Werkstück eine gute Oberflächengüte bei hoher Genauigkeit im µm-Bereich bei extrem großen Bearbeitungsvorschüben >10 m/min aufweisen.

Durch verschiedene Bearbeitungsstrategien wird mit Hilfe des CYCLE832 das NC-Programm sehr genau abgestimmt:

- Bei Schruppbearbeitung liegt durch Überschleifen der Kontur die Gewichtung auf der Geschwindigkeit.
- Bei der Schlichtbearbeitung liegt die Gewichtung auf der Genauigkeit.

In beiden Fällen wird durch Angabe einer Toleranz die Bearbeitungskontur eingehalten, um die gewünschte Oberflächengüte zu erreichen.

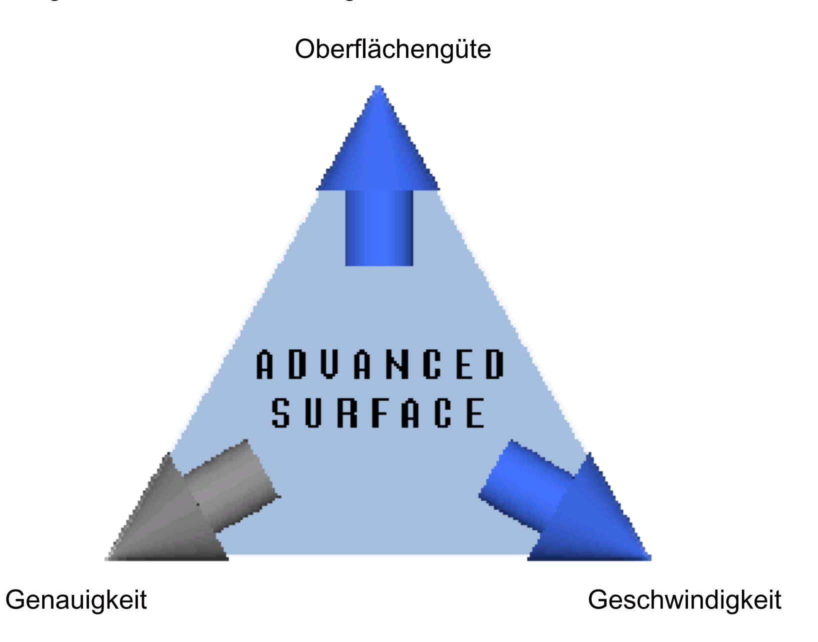


Bild 5-2 Bearbeitung der Werkstückoberfläche

Einstellungen für Advanced Surface prüfen

Nach Installation der Toolbox 828D sind im Verzeichnis \Examples\AS_checkprograms folgende Prüfprogramme vorhanden:

- MDC3AXV3B_SW2_6.SPF
- MDC3AXV3A_SW2_7__SW4_4.SPF
- MDC3AXV3B_SW4_5.SPF

Zusätzlich enthält die Datei readme.txt weitere Informationen, um mit den Programmen die empfohlenen Einstellungen der Maschinendaten für Advanced Surface zu prüfen. Beim Starten des Programms wird im Verzeichnis SPF.DIR eine Datei MDADVS.SPF erzeugt, in der die Ergebnisse der Prüfung als Liste ausgegeben werden.

Beispiel:

(nur Auszug aus der kompletten Datei)

```
Machine Data setting different to required Advanced Surface Setting
Machine Data setting different to recommended Advanced Surface Setting
N20470 $MC_CPREC_WITH_FFW=1, recommended: 3
N20485 $MC_COMPRESS_SMOOTH_FACTOR[2]=0, recommended: 0.0001
N20485 $MC_COMPRESS_SMOOTH_FACTOR[3]=0, recommended: 0.0001
N20485 $MC_COMPRESS_SMOOTH_FACTOR[4]=0, recommended: 0.0001
N20486 $MC_COMPRESS_SPLINE_DEGREE[2]=3, recommended: 5
N20486 $MC_COMPRESS_SPLINE_DEGREE[3]=3, recommended: 5
N20486 $MC_COMPRESS_SPLINE_DEGREE[4]=3, recommended: 5
N20487 $MC_COMPRESS_SMOOTH_FACTOR_2[2]=0, recommended: 0.5
N20487 $MC_COMPRESS_SMOOTH_FACTOR_2[3]=0, recommended: 0.5
N20487 $MC_COMPRESS_SMOOTH_FACTOR_2[4]=0, recommended: 0.5
```

5.4 Maschinendaten für Advanced Surface

Maschinendaten für Advanced Surface

Um eine optimale Werkstückoberfläche bei der Bearbeitung von Formenbauwerkstücken zu erzielen, werden folgende Sollwerte für die aufgelisteten Maschinendaten empfohlen:

Allgemeine Maschinendaten:

MD-Nummer	Bezeichnung	Vorein-stellung	Empfehlung	Daten-klasse	Parameter
10200	$MN_INT_INCR_PER_MM	100000	100000	M	--
10210	$MN_INT_INCR_PER_DEG	100000	100000	M	--
10682	$MN_CONTOUR_SAMPLING_FACTOR	1	1	M	--

Kanalspezifische Maschinendaten:

MD-Nummer	Bezeichnung	Vorein-stellung	Empfehlung	Daten-klasse	Parameter
20150[3]	$MC_GCODE_RESET_VALUES	3	3	M	FIFOCTRL
20150[44]	$MC_GCODE_RESET_VALUES	2	2	M	UPATH
20170	$MC_COMPRESS_BLOCK_PATH_LIMIT	20	20	M	--
20172	$MC_COMPRESS_VELO_TOL	60000	1000	M	--
20443[0]	$MC_LOOKAH_FFORM	0	0	M	DYNNORM
20443[1]	$MC_LOOKAH_FFORM	0	0	M	DYNPOS
20443[2]	$MC_LOOKAH_FFORM	0	1	M	DYNROUGH
20443[3]	$MC_LOOKAH_FFORM	0	1	M	DYNSEMIFIN
20443[4]	$MC_LOOKAH_FFORM	0	1	M	DYNFINISH
20455	$MC_LOOKAH_FUNCTION_MASK	0	0	M	--
20480	$MC_SMOOTHING_MODE	0	0	M	--
20482	$MC_COMPRESSOR_MODE	100	100	M	--
20485	$MC_COMPRESS_SMOOTH_FACTOR[0-4]	0	0	M	--
20486	$MC_COMPRESS_SPLINE_DEGREE[0-4]	3	3	M	--
20490	$MC_IGNORE_OVL_FACTOR_FOR_ADIS	1	1	M	--
20550	$MC_EXACT_POS_MODE	3	3	M	--
20600[0]	$MC_MAX_PATH_JERK	10000	10000	M	DYNNORM
20600[1]	$MC_MAX_PATH_JERK	10000	10000	M	DYNPOS
20600[2]	$MC_MAX_PATH_JERK	10000	10000	M	DYNROUGH
20600[3]	$MC_MAX_PATH_JERK	10000	10000	M	DYNSEMIFIN
20600[4]	$MC_MAX_PATH_JERK	10000	10000	M	DYNFINISH
20602[0]	$MC_CURV_EFFECT_ON_PATH_ACCEL	0	0	M	--
20602[1]	$MC_CURV_EFFECT_ON_PATH_ACCEL	0	0	M	--
20602[2]	$MC_CURV_EFFECT_ON_PATH_ACCEL	0,6	0,65	M	--

MD-Nummer	Bezeichnung	Vorein-stellung	Empfehlung	Daten-klasse	Parameter
20602[3]	$MC_CURV_EFFECT_ON_PATH_ACCEL	0,6	0,6	M	--
20602[4]	$MC_CURV_EFFECT_ON_PATH_ACCEL	0,6	0,5	M	--
20603[0]	$MC_CURV_EFFECT_ON_PATH_JERK	0	0	M	--
20603[1]	$MC_CURV_EFFECT_ON_PATH_JERK	0	0	M	--
20603[2]	$MC_CURV_EFFECT_ON_PATH_JERK	0	0	M	--
20603[3]	$MC_CURV_EFFECT_ON_PATH_JERK	0	0	M	--
20603[4]	$MC_CURV_EFFECT_ON_PATH_JERK	0	0	M	--
20605[0]	$MC_PREPDYN_SMOOTHING_FACTOR	0	1	M	--
20605[1]	$MC_PREPDYN_SMOOTHING_FACTOR	0	1	M	--
20605[2]	$MC_PREPDYN_SMOOTHING_FACTOR	0	1	M	--
20605[3]	$MC_PREPDYN_SMOOTHING_FACTOR	0	1	M	--
20605[4]	$MC_PREPDYN_SMOOTHING_FACTOR	0	1	M	--
20606[0]	$MC_PREPDYN_SMOOTHING_ON	0	0	M	DYNNORM
20606[1]	$MC_PREPDYN_SMOOTHING_ON	0	0	M	DYNPOS
20606[2]	$MC_PREPDYN_SMOOTHING_ON	0	1	M	DYNROUGH
20606[3]	$MC_PREPDYN_SMOOTHING_ON	0	1	M	DYNSEMIFIN
20606[4]	$MC_PREPDYN_SMOOTHING_ON	0	1	M	DYNFINISH
28060	$MC_MM_IPO_BUFFER_SIZE	150	150	M	--
28070	$MC_MM_NUM_BLOCKS_IN_PREP	80	80	M	--
28302[0]	$MC_MM_PROTOC_NUM_ETP_STD_TYP	28	28	M	--
28520	$MC_MM_MAX_AXISPOLY_PER_BLOCK	3	5	M	--
28530	$MC_MM_PATH_VELO_SEGMENTS	5	5	M	--
28533	$MC_MM_LOOKAH_FFORM_UNITS	18	18	M	--
28540	$MC_MM_ARCLENGTH_SEGMENTS	10	10	M	--

Achs-Maschinendaten:

MD-Nummer	Bezeichnung	Vorein-stellung	Empfehlung	Daten-klasse	Parameter
32300	$MA_MAX_ACCEL[0-3]	1	1	M	--
32300	$MA_MAX_ACCEL[4]	1	1	M	--
32310[0]	$MA_MAX_ACCEL_OVL_FACTOR	1.2	1.2	M	--
32310[1]	$MA_MAX_ACCEL_OVL_FACTOR	1.2	1.2	M	--
32310[2]	$MA_MAX_ACCEL_OVL_FACTOR	1.2	1.2	M	--
32310[3]	$MA_MAX_ACCEL_OVL_FACTOR	1.2	1.2	M	--
32310[4]	$MA_MAX_ACCEL_OVL_FACTOR	1.2	1.2	M	--
32400	$MA_AX_JERK_ENABLE	0	0	M	--
32402	$MA_AX_JERK_MODE	2	2	M	--
32410	$MA_AX_JERK_TIME	0,001	0,001	M	--
32431[0]	$MA_MAX_AX_JERK	1000000	1000000	I	DYNNORM

MD-Nummer	Bezeichnung	Voreinstellung	Empfehlung	Datenklasse	Parameter
32431[1]	$MA_MAX_AX_JERK	1000000	1000000	I	DYNPOS
32431[2]	$MA_MAX_AX_JERK	1000000	1000000	I	DYNROUGH
32431[3]	$MA_MAX_AX_JERK	20	20	I	DYNSEMIFIN
32431[4]	$MA_MAX_AX_JERK	20	20	I	DYNFINISH
32432[0]	$MA_PATH_TRANS_JERK_LIM	1000000	1000000	I	DYNNORM
32432[1]	$MA_PATH_TRANS_JERK_LIM	1000000	1000000	I	DYNPOS
32432[2]	$MA_PATH_TRANS_JERK_LIM	1000000	1000000	I	DYNROUGH
32432[3]	$MA_PATH_TRANS_JERK_LIM	20	20	I	DYNSEMIFIN
32432[4]	$MA_PATH_TRANS_JERK_LIM	20	20	I	DYNFINISH
32620	$MA_FFW_MODE	3	3	M	DYNNORM
32640	$MA_STIFFNESS_CONTROL_ENABLE	1	1	M	DSC
33120	$MA_PATH_TRANS_POS_TOL	0,005	0,005	U	--

Setting-Datum:

MD-Nummer	Bezeichnung	Voreinstellung	Empfehlung	Datenklasse	Parameter
42470	$SC_CRIT_SPLINE_ANGLE	36°	36°	U	--
42471	$SC_MIN_CURV_RADIUS	1	1	M	--
42500	$SC_SD_MAX_PATH_ACCEL	10000	10000	U	--
42502	$SC_IS_SD_MAX_PATH_ACCEL	0	0	U	--
42510	$SC_SD_MAX_PATH_JERK	10000	10000	U	--
42512	$SC_IS_SD_MAX_PATH_JERK	0	0	U	--

5.5 Maschinendaten für eine Analog-Spindel

Anschluss für Geber-Istwerte

Voraussetzung für den Anschluss eines Spindel-Istwertgebers ist eine Achse mit digitalem Antrieb (SINAMICS-Achse), die nur einen Geber hat. Als zweiter Geber wird an das Achsmodul ein Sensor Module z. B. SMC30 angeschlossen, das mit dem direkt angebauten Spindel-Istwertgeber verbunden ist. Das PROFIdrive-Telegramm der SINAMICS-Achse muss für zwei Geber-Istwerte ausgelegt sein. Die Standard-SDB der SINUMERIK 828D unterstützen abhängig von MD11240 $MN_PROFIBUS_SDB_NUMBER Siemens-Telegramm 116 oder Siemens-Telegramm 136.

Funktionale Einschränkungen:

- Da es bei der Analog-Spindel kein Sollwert-Telegramm gibt, ist darauf aufbauende Funktionalität nicht vorhanden, z. B. DSC, Momentenvorsteuerung.
- Das Signal "Drive Ready" wird von der Treibersoftware als gesetzt simuliert.
- Da auf Antriebsparameter nicht zugegriffen werden kann, ist davon betroffene Funktionalität eingeschränkt (z. B. Spindelauslastungsanzeige, Automatische Drehzahl- und Lageregleroptimierung). In der Anzeige "Service Übersicht" werden in der Spalte der Analog-Spindel die fehlenden Werte grau gekennzeichnet: "Diese Achse ist nicht betroffen."

Anschluss für Spindel-Sollwerte

Die analoge Spindel wird an die Schnittstelle X252 angeschlossen. In Abhängigkeit von der Betriebsart der analogen Spindel werden folgende Signale ausgegeben:

Spindeltyp	Signalname	Bedeutung
Bipolare Spindel	AOUT	Analogausgang +/-10 V
	AGND	Analogausgang 0 V Bezugssignal
	DOUT11	Reglerfreigabe
Unipolare Spindel mit getrennten Freigabe- und Richtungssignalen	AOUT	Analogausgang +10 V
	AGND	Analogausgang 0 V Bezugssignal
	DOUT12	negative Fahrrichtung (bleibt auch bei fehlender Reglerfreigabe gesetzt)
unipolare Spindel mit richtungsabhängiger Freigabe	AOUT	Analogausgang +10 V
	AGND	Analogausgang 0 V Bezugssignal
	DOUT11	Reglerfreigabe und positive Fahrrichtung
	DOUT12	Reglerfreigabe und negative Fahrrichtung

Die D/A-Wandlung des Spindelsollwerts erfolgt mit einer Auflösung von 14 Bit.

Relevante Maschinendaten

Folgende Maschinendaten sind für die analoge Spindel einzustellen:

- Aktivierung der analogen Spindel:

  MD30100:$MA_CTRLOUT_SEGMENT_NR = 0

  Durch das Einstellen des Lokalbus als Bussegment wird die analoge Spindel für die Sollwertausgabe aktiviert.

- Betriebsart der analogen Spindel auswählen:

MD30134:$MA_IS_UNIPOLAR_OUTPUT

=0	bipolarer Ausgang (+/-10 V)
=1	unipolare Spindel mit getrennten Freigabe- und Richtungssignalen
=2	unipolare Spindel mit richtungsabhängiger Freigabe

- Die Nennausgangsspannung muss an die Nenndrehzahl des Analogantriebes angepasst werden:

  MD32250: $MA_RATED_OUTVAL

  MD32260: $MA_RATED_VELO

- Soll die analoge Spindel ohne Istwertgeber betrieben werden, ist als Anzahl der Geber der Wert 0 einzutragen:

  MD30200: $MA_NUM_ENCS = 0

- Beim Einsatz eines direkt angebauten Spindel-Istwertgebers ist der Typ der Istwerterfassung PROFIBUS auszuwählen:

  MD30240: $MA_ENC_TYPE = 1 oder 4

- Der Spindel-Istwertgeber muss als 2. Geber einer SINAMICS Achse konfiguriert werden. Dazu ist die Antriebszuordnung des Istwertes so wie bei der Achse einzustellen, an deren Achsmodul das SMC30 angeschlossen ist:

  MD30220[0]: $MA_ENC_MODULE_NR[0] = MD30220[0] der SINAMICS Achse

- Die Eingangszuordnung des Istwertes ist auf den Eingang für den 2. Geber am SINAMICS Achsmodul zu setzen:

  MD30230[0]: $MA_ENC_INPUT_NR[0] = 2

- Eine automatische Driftkompensation kann bei angeschlossenem Spindel-Istwertgeber aktiviert werden:

  MD36700:$MA_DRIFT_ENABLE

  MD36710: $MA_DRIFT_LIMIT

- Ein Drift-Grundwert wird unabhängig vom Vorhandensein eines Spindel-Istwertgebers als zusätzlicher Drehzahlsollwert ständig aufgeschaltet:

  MD36720:$MA_DRIFT_VALUE

Beispiel

Beispiel für 3 Achsen und analoge Spindel mit Istwertgeber (SMC30 als zweiter Geber an Y-Achse angeschlossen).

MD-Bezeichner	X	Y	Z	SP	Bedeutung
MD30100 CTRLOUT_SEGMENT_NR	5	5	5	0	Bussegment
MD30110 CTRLOUT_MODULE_NR	1	2	3	1	Zuordnung Baugruppe
MD30120 CTRLOUT_NR	1	1	1	1	Zuordnung Ausgang
MD30130 CTRLOUT_TYPE	1	1	1	1	Ausgabeart
MD30134 IS_UNIPOLAR_OUTPUT	0	0	0	0	"0" bedeutet bipolar
MD30200 NUM_ENCS	1	1	1	1	Geberanzahl
MD30220[0] ENC_MODULE_NR[0]	1	2	3	2	SMC30 an Y-Achse
MD30230[0] ENC_INPUT_NR[0]	1	1	1	2	Eingang für zweiten Geber
MD30240 ENC_TYPE	1/4	1/4	1/4	1/4	Geberart

MD-Bezeichner	SP	Bedeutung
MD32250 RATED_OUTVAL	80 (80% von 10V)	Nennausgangsspannung
MD32260 RATED_VELO	3000 (bei 8V)	Motornenndrehzahl
MD36700 DRIFT_ENABLE	0	Automatische Driftkompensation ist deaktiviert.
MD36710 DRIFT_LIMIT	0	Begrenzung für automatische Driftkompensation
MD36720 DRIFT_VALUE	0	Drift-Grundwert

5.6 Daten verwalten

Anwendung

Die Funktion "Daten verwalten" dient zur Unterstützung und Vereinfachung der Inbetriebnahme und stellt Funktionen zum Sichern, Laden und Vergleichen folgender Daten zur Verfügung:

- Maschinendaten
- Setting-Daten
- Antriebsdaten
- Kompensationsdaten

Im Gegensatz zu einem Inbetriebnahmearchiv wird hier nur ein einzelnes Steuerungsobjekt (Achse, Kanal, SERVO, Einspeisung usw.) im ASCII-Format (*.tea) gespeichert. Diese Datei kann editiert und auf andere Steuerungsobjekte vom gleichen Typ übertragen werden. Die Funktion "Daten verwalten" dient auch für das Übertragen von Antriebsobjekten bei SINAMICS-Antrieben.

Daten verwalten

Mit der Funktion "Daten verwalten" stehen folgende Möglichkeiten zur Verfügung:

- Daten innerhalb der Steuerung übertragen
- Daten in eine Datei speichern
- Daten in eine Datei laden
- Daten vergleichen

Folgende Daten können gesichert werden und werden unter folgendem absoluten Pfad auf der CompactFlash Card abgelegt:

- user/sinumerik/hmi/data/backup/ec für Kompensationsdaten
- user/sinumerik/hmi/data/backup/md für Maschinendaten
- user/sinumerik/hmi/data/backup/sd für Setting-Daten
- user/sinumerik/hmi/data/backup/snx für SINAMICS-Parameter

Die Funktion bedienen Sie unter "Inbetriebnahme" → "Maschinendaten" → "Daten verwalten".

Beispiel für "Daten innerhalb der Steuerung übertragen"

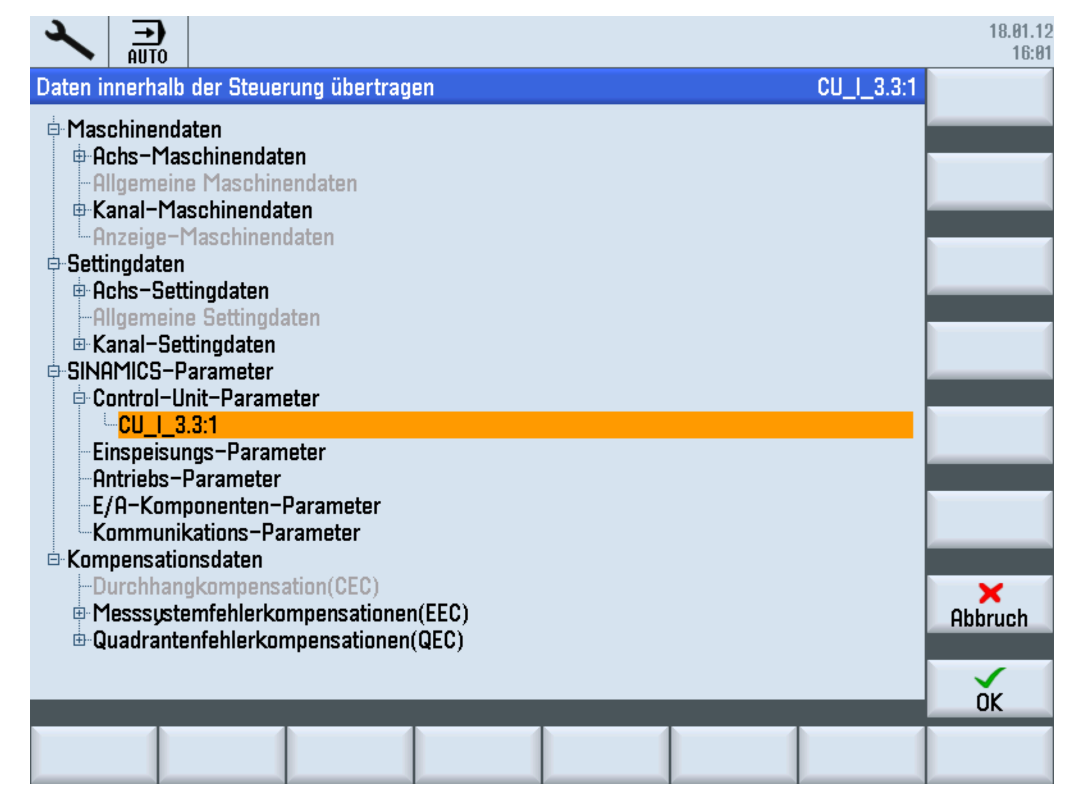


Bild 5-3 Daten verwalten

5.6.1 So übertragen Sie Daten innerhalb der Steuerung

Daten innerhalb der Steuerung übertragen

Hinweis

Schutz der Maschine

Aus Sicherheitsgründen sollte die Übertragung von Maschinendaten und Setting-Daten nur bei einer gesperrten Freigabe erfolgen.

Vorgehensweise:

1. Wählen Sie die Option "Daten innerhalb der Steuerung übertragen".
2. In der Datenstruktur wählen Sie die Quelldaten aus und Bestätigen Sie mit "OK".
3. Wählen Sie in der Auswahlliste ein Objekt aus, z. B. eine andere Achse oder ein anderes Antriebsobjekt, auf das Sie die Daten übertragen wollen und bestätigen Sie mit "OK".

4. Beachten Sie die Sicherheitshinweise und überprüfen Sie die Freigaben an der Maschine sowie des Antriebs.
5. Bei Antriebsdaten werden die Daten mit dem Softkey "Laden" auf das Zielobjekt übertragen.

5.6.2 So speichern und laden Sie Daten

Daten in eine Datei speichern

Vorgehensweise:

1. Wählen Sie die Option "Daten in eine Datei speichern".
2. In der Datenstruktur wählen Sie die Daten, die Sie in eine Datei speichern wollen und bestätigen mit "OK".
3. Als Speicherort wählen Sie ein Verzeichnis oder ein USB-Speichermedium und geben einen Namen ein.

Hinweis

SINAMICS-Parameter

Beim Speichern wird immer eine ASCII Datei (*.TEA) erzeugt.

Beim Speichern von Antriebsdaten werden drei Dateien folgenden Typs erzeugt:

- eine Binär-Datei (*.ACX), die nicht lesbar ist.
- eine ASCII Datei (*.TEA), die im ASCII-Editor bearbeitet oder gelesen werden kann.
- eine Protokolldatei (*.log), die im Fehlerfall Meldetexte enthält oder bei erfolgreicher Speicherung leer ist.

Daten aus einer Datei laden

Hinweis

Schutz der Maschine

Aus Sicherheitsgründen sollte die Übertragung von Maschinendaten und Setting-Daten nur bei gesperrter Freigabe erfolgen.

Vorgehensweise:

1. Wählen Sie die Option "Daten aus einer Datei laden".
2. In der Datenstruktur wählen Sie die gespeicherte Datei und bestätigen mit "OK".
3. Wählen Sie in der Auswahlliste ein Objekt aus, z. B. eine andere Achse oder ein anderes Antriebsobjekt, auf das Sie die Daten übertragen wollen und bestätigen Sie mit "OK".

4. Beachten Sie die Sicherheitshinweise und überprüfen Sie die Freigaben an der Maschine sowie des Antriebs.
5. Bei Antriebsdaten werden die Daten mit dem Softkey "Laden" auf das Zielobjekt übertragen.

5.6.3 So vergleichen Sie Daten

Daten vergleichen

Beim Daten vergleichen können Sie unterschiedlichen Datenquellen auswählen: die aktuellen Daten auf der Steuerung oder Daten, die in Dateien gespeichert sind.

Vorgehensweise:

1. Wählen Sie die Option "Daten vergleichen".
2. In der Datenstruktur wählen Sie die Daten aus, die Sie vergleichen wollen.
3. Übernehmen Sie Daten in die Liste im unteren Anzeigebereich mit dem Softkey "In Liste aufnehmen".
4. Entfernen Sie Daten wieder mit dem Softkey "Aus Liste löschen".
5. Sind mehr als 2 Datenobjekte in der Liste, können Sie durch Aktivieren des Kontrollkästchens mehrere oder 2 Datenobjekte aus der Liste vergleichen.
6. Starten Sie den Vergleich mit dem Softkey "Vergleichen". Bei umfangreichen Parameterlisten kann die Anzeige des Vergleichsergebnisses einige Zeit dauern.
7. Mit dem Softkey "Legende" blenden Sie die Legende ein und aus. Voreingestellt ist folgende Anzeige:
   – Unterschiedliche Parameter werden angezeigt.
   – Gleiche Parameter werden nicht angezeigt.
   – Parameter, die nicht überall vorhanden sind, werden angezeigt.

6 Antrieb in Betrieb nehmen

6.1 Antrieb konfigurieren

6.1.1 Beispiel: Konfiguration mit SINAMICS S120 Combi

Übersicht

Die Inbetriebnahme-Software für SINAMICS S120 wird kostenlos auf der Toolbox CD zur Verfügung gestellt. Bis zur vollständigen Verfügbarkeit der SINAMICS S120 Inbetriebnahme-Funktionalität in der Bedienoberfläche, wird hier die Inbetriebnahme des Antriebs in zwei Schritten beschrieben:

- Über die Funktionalität in der Bedienoberfläche wird die Topologie automatisch erkannt und bereitgestellt.
- Die Konfiguration und Parametrierung der Geber erfolgt mit der Inbetriebnahme-Software für SINAMICS S120. Die Verbindung des PG/PC wird über die frontseitige Ethernet-Schnittstelle der SINUMERIK 828D hergestellt.

Konfiguration des Antriebs

Für die Konfiguration des Antriebs wird das Beispiel 2 mit SINAMICS S120 Combi aus Kapitel "Systemübersicht (Seite 11)" gewählt. Die DRIVE-CLiQ-Verbindungen müssen den Topologieregeln in Kapitel "Topologieregeln für S120 Combi (Seite 166)" entsprechen.

Die Reihenfolge der DRIVE-CLiQ-Verbindungen entspricht der Reihenfolge der SINAMICS Antriebsobjekt-Nummer (= Voreinstellung):

Achsen	SINAMICS Antriebsobjekt	
	Nummer	Name
---	1	Control Unit
---	2	Line Module
MSP1	3	SERVO 1
MX1	4	SERVO 2
MY1	5	SERVO 3
MZ1	6	SERVO 4
MA1	7	SERVO 5
---	11	HUB Module
---	9	TM54F Master

Ablauf

Der Ablauf ist in folgende Schritte aufgeteilt:

- Schritt 1: Spindel konfigurieren.
- Schritt 2: Achsen konfigurieren.
- Schritt 3: Achsen zuordnen und Daten sichern:

  Vorgehensweise entsprechend Kapitel "Beispiel: So ordnen Sie die Achsen zu (Seite 147)".

Die Schritte sind detailliert in den nachfolgenden Kapiteln beschrieben.

Ausgangszustand

Bevor Sie beginnen:

- PG/PC mit der Steuerung verbinden: siehe Kapitel "Kommunikation mit der Steuerung (Seite 21)"
- Der Hochlauf der Steuerung wird mit "Siemens default data" durchgeführt.
- Anzeige im Bedienbereich "Inbetriebnahme" an der Steuerung:

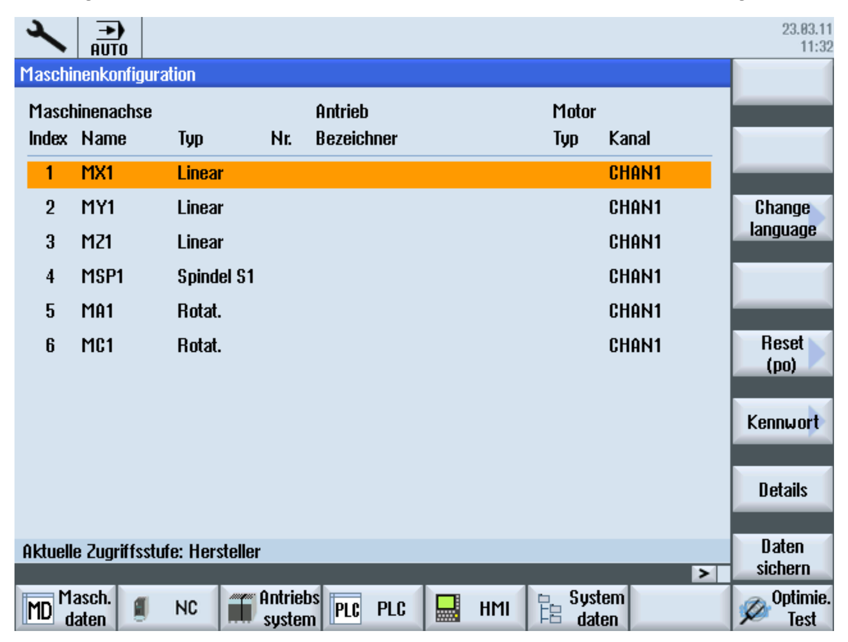


Bild 6-1 Steuerung nach Hochlauf mit "Siemens default data"

Vorgehensweise

Um den Antrieb zu konfigurieren:

1. Starten Sie mit dem Softkey "Antriebssystem" die Inbetriebnahme.
2. Bestätigen Sie die folgende Frage mit "OK", um die Gerätekonfiguration zu starten.

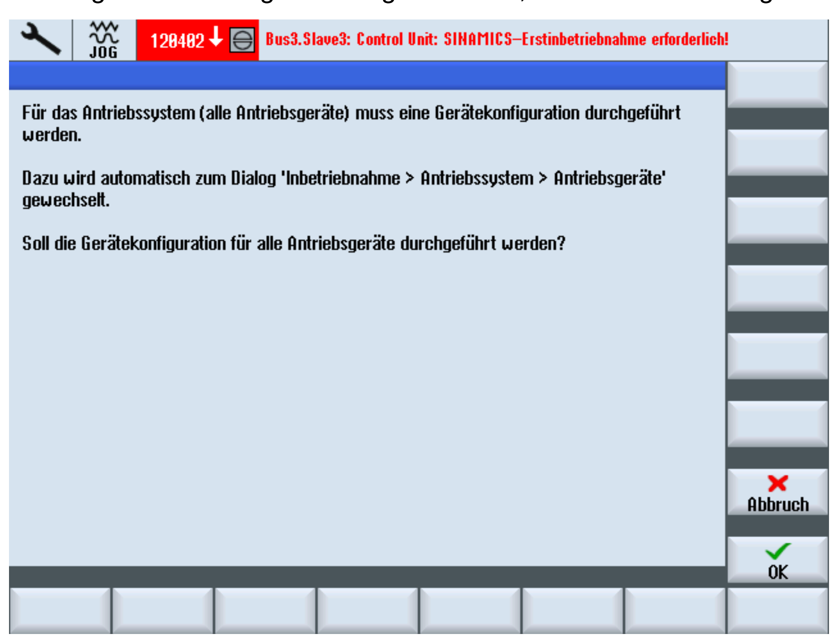


Bild 6-2 Start Inbetriebnahme mit SINUMERIK Operate

Danach wird die Topologie des Systems automatisch ausgelesen.

3. Bestätigen Sie die folgende Frage mit "OK", um die Gerätekonfiguration durchzuführen.

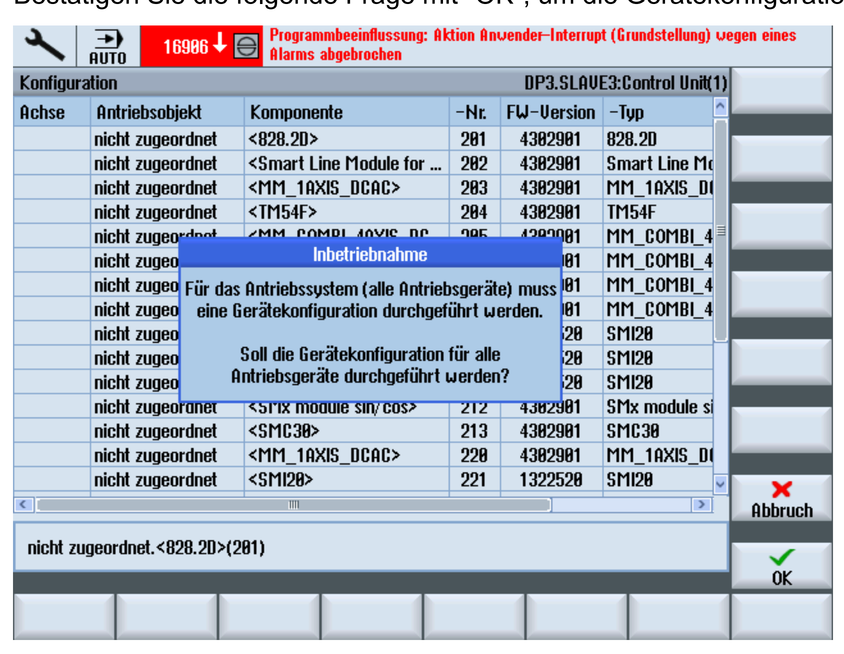


Bild 6-3 Gerätekonfiguration

4. Bestätigen Sie die folgende Frage mit "OK", um einen NCK Power-On-Reset durchzuführen. Dies kann einige Minuten dauern.

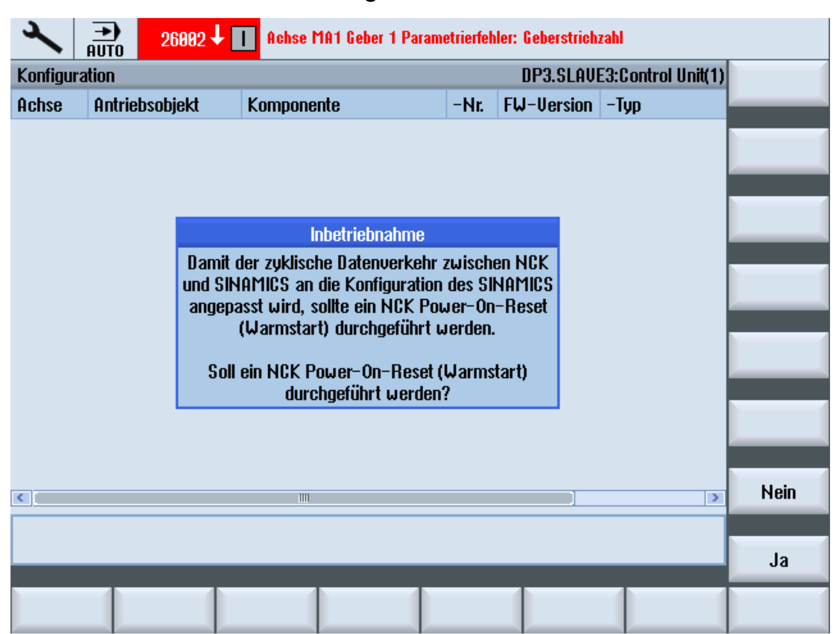


Bild 6-4 Power-On-Reset bestätigen

5. Nachdem dem Hochlauf erhalten Sie folgende Meldung:
6. Bestätigen Sie mit "OK".

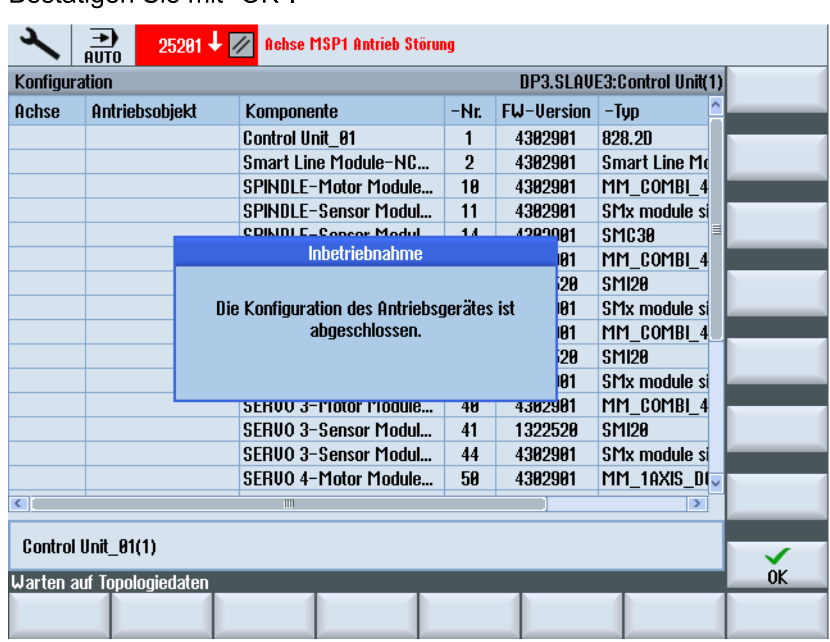


Bild 6-5 Abschluss der automatischen Konfiguration

7. Nachdem die Topologiedaten ermittelt wurden, ist die Konfiguration abgeschlossen.

   Für die weitere Inbetriebnahme wird die Inbetriebnahme-Software für SINAMICS S120 (Seite 13) auf PG/PC benötigt.

8. Wählen Sie den Bedienbereich "Inbetriebnahme".
9. Setzen Sie das Kennwort auf Zugriffsstufe "Hersteller".
10. Bestätigen Sie die folgende Frage mit "OK", um die weitere Konfiguration zu starten.

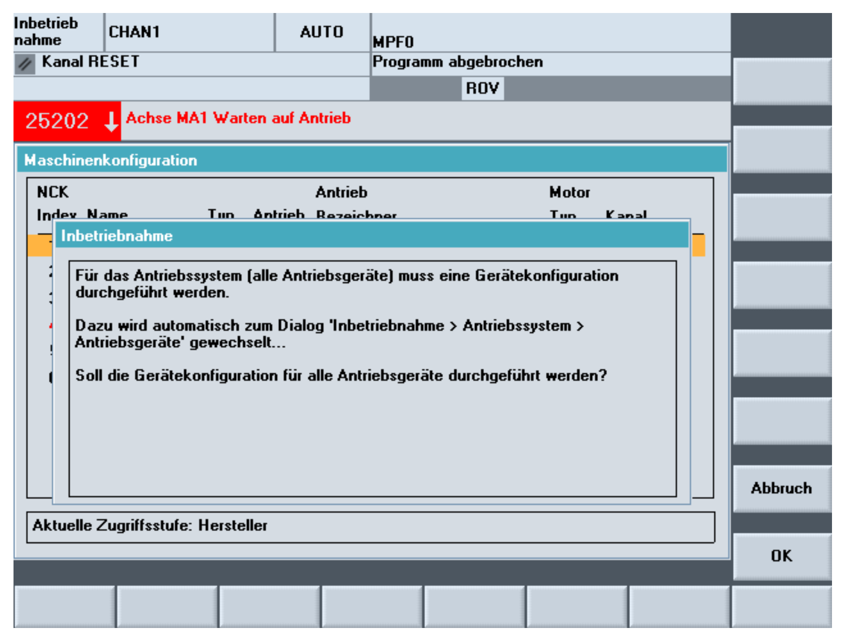


Bild 6-6 Start Inbetriebnahme

6.1.2 Beispiel: So konfigurieren Sie eine Spindel mit SMC-Geber

Spindel-Konfiguration mit SMC20

Vorgehensweise:

1. Wählen Sie zuerst mit dem Softkey "Antrieb +" oder "Antrieb -" die Spindel aus.
2. Um die Spindel zu konfigurieren, wählen Sie "Ändern".

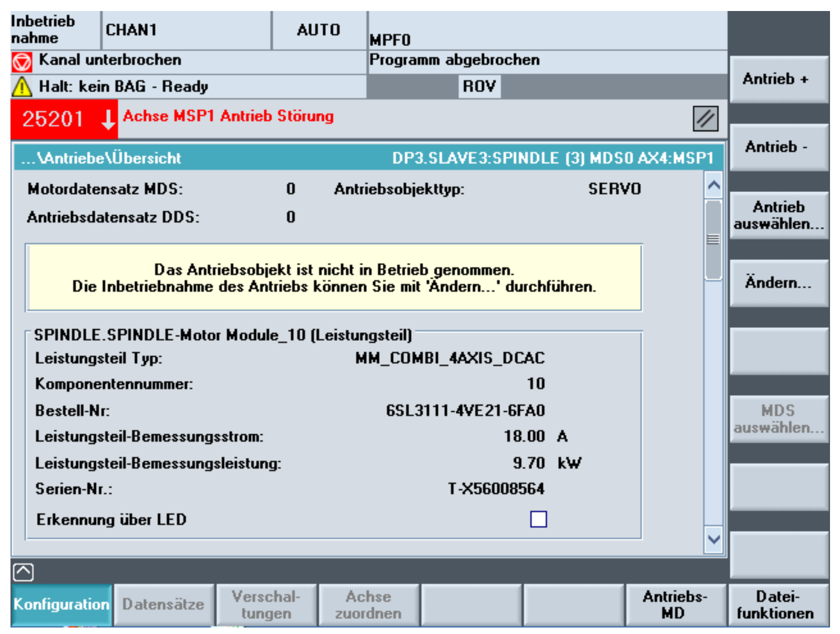


Bild 6-7 Antriebe\Übersicht

Es werden folgende Spindeldaten angezeigt:

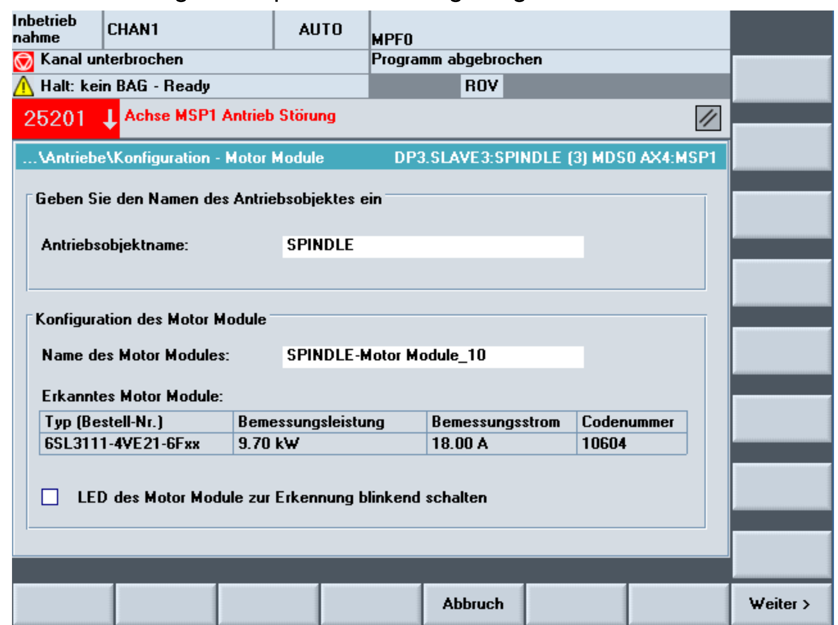


Bild 6-8 Antriebe\Konfiguration: Motor Module (Spindel)

3. Wählen Sie "Weiter >", um den zugehörigen Motor zuzuordnen: Der Motor kann über den "Motortyp" oder über die "Suche" der Codenummer gewählt werden.

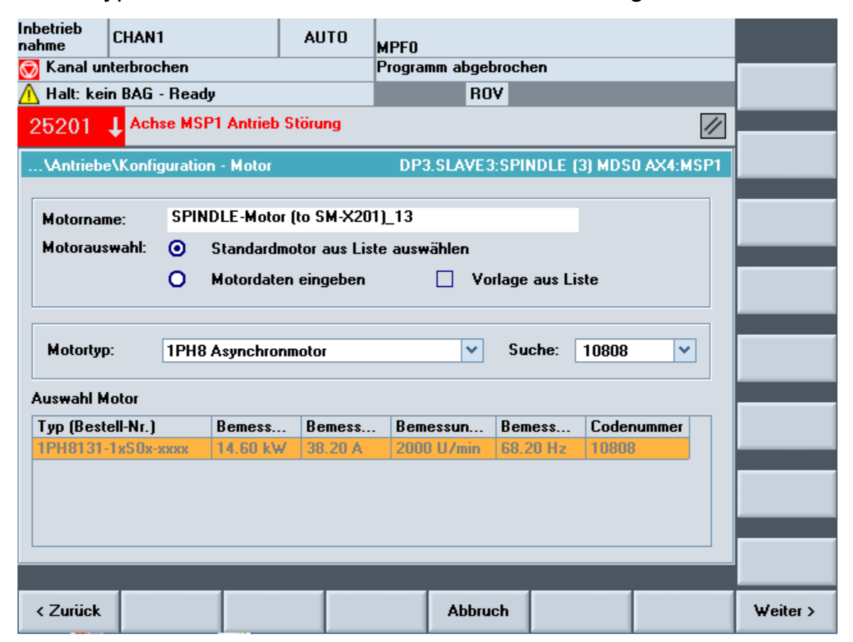


Bild 6-9 Antriebe\Konfiguration: Motor

4. Wählen Sie "Weiter >", um die Motorhaltebremse zu konfigurieren:

   Beachten Sie den Hinweis zur Lagerausführung.

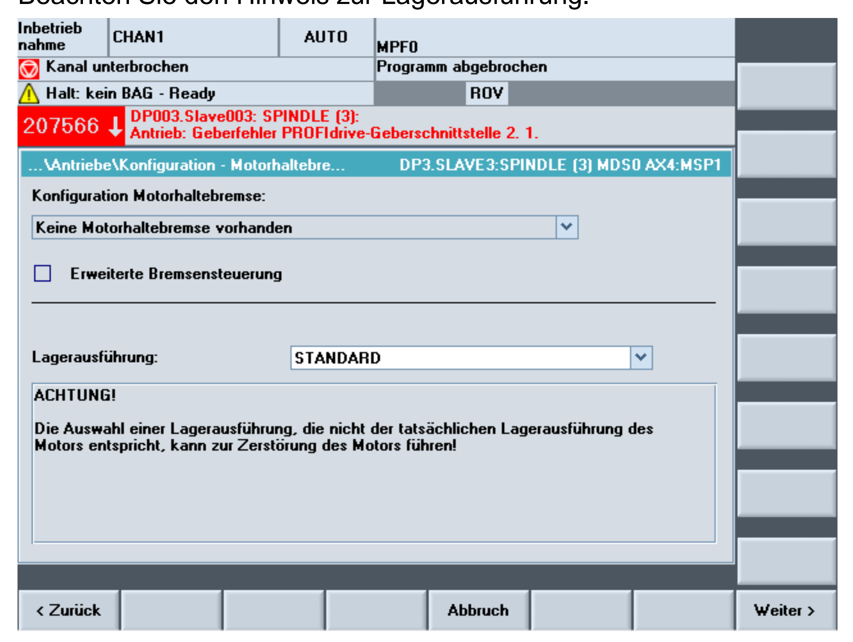


Bild 6-10 Antriebe\Konfiguration: Motorhaltebremse

5. Wählen Sie "Weiter >", um die Geber zuzuordnen: die beiden Geber (am Motormess-System und am direkten Mess-System) wurden bereits korrekt bei der automatischen Ermittlung der Topologie zugeordnet.

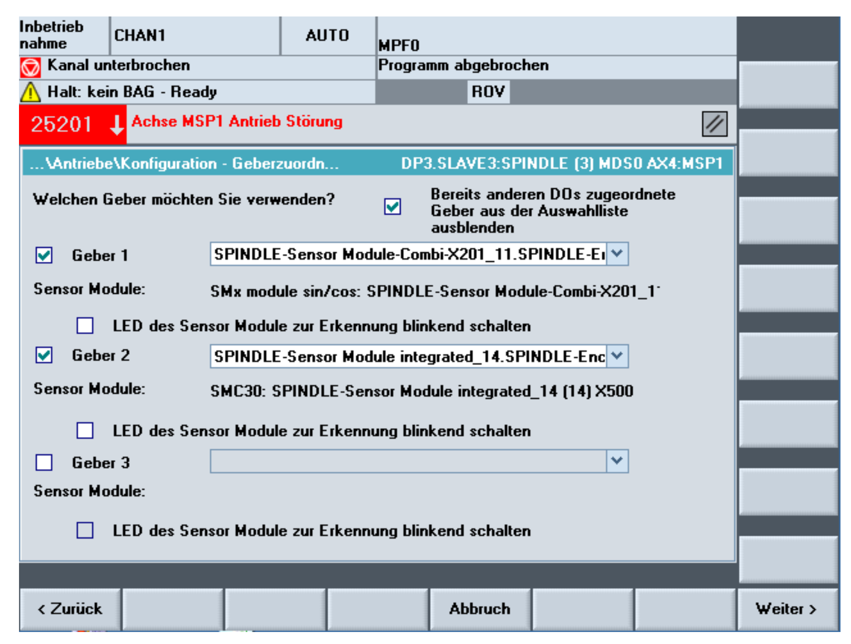


Bild 6-11 Antriebe\Konfiguration: Geberzuordnung

6. Wählen Sie "Weiter >", um den Motorgeber zu konfigurieren.

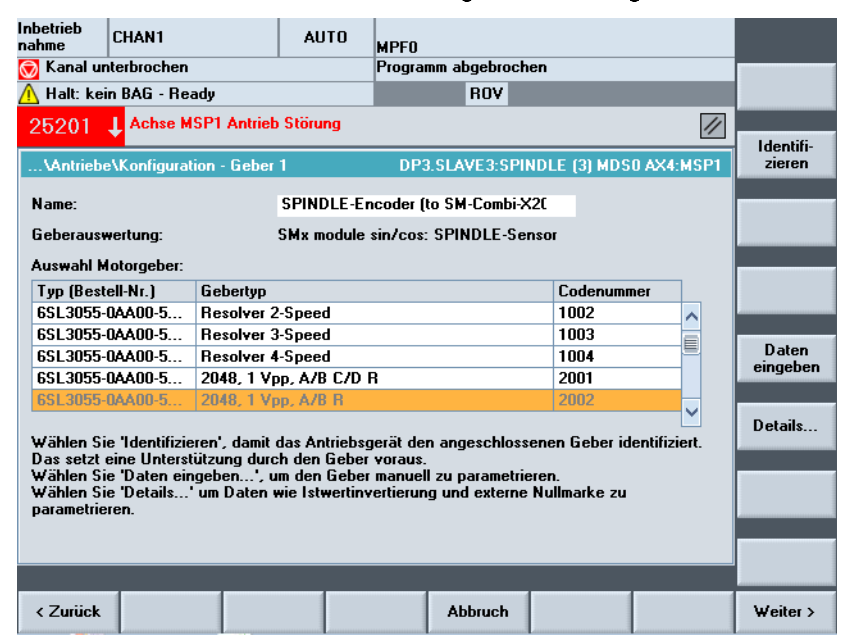


Bild 6-12 Antriebe\Konfiguration - Geber 1

7. Wählen Sie "Daten eingeben", um die Eigenschaften des Gebers auf den Wert von 4096 Umdrehungen anzupassen.

8. Wenn Sie diese Einstellung mit "OK" bestätigen, wird der Geber als "Benutzerdefiniert" angezeigt:

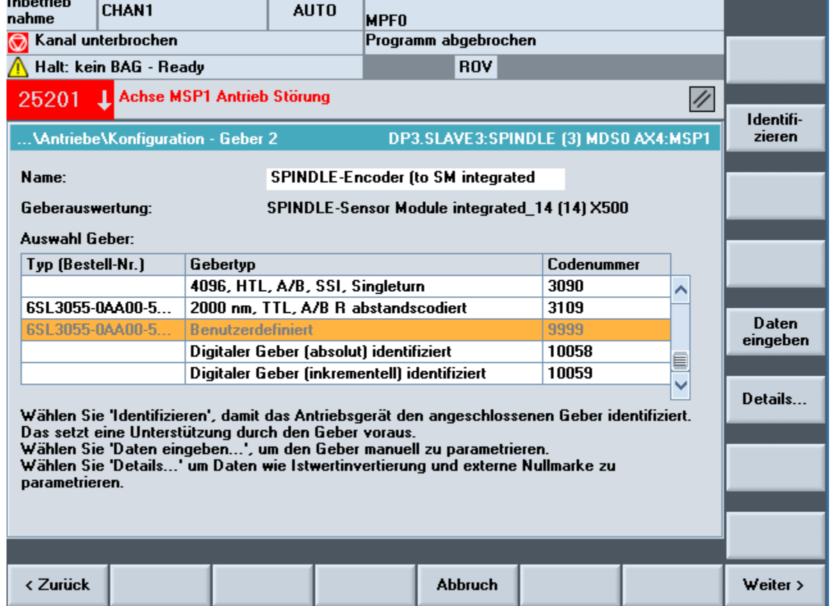


Bild 6-13 Antriebe\Konfiguration - Geber 2: benutzerdefiniert

9. Wählen Sie "Weiter >", um die Voreinstellung mit Telegramm 116 (Drehzahlvorsteuerung) zu übernehmen. Damit werden für die internen Antriebsgrößen zusätzliche Prozessdaten (PZD) übertragen und in Systemvariablen abgelegt.

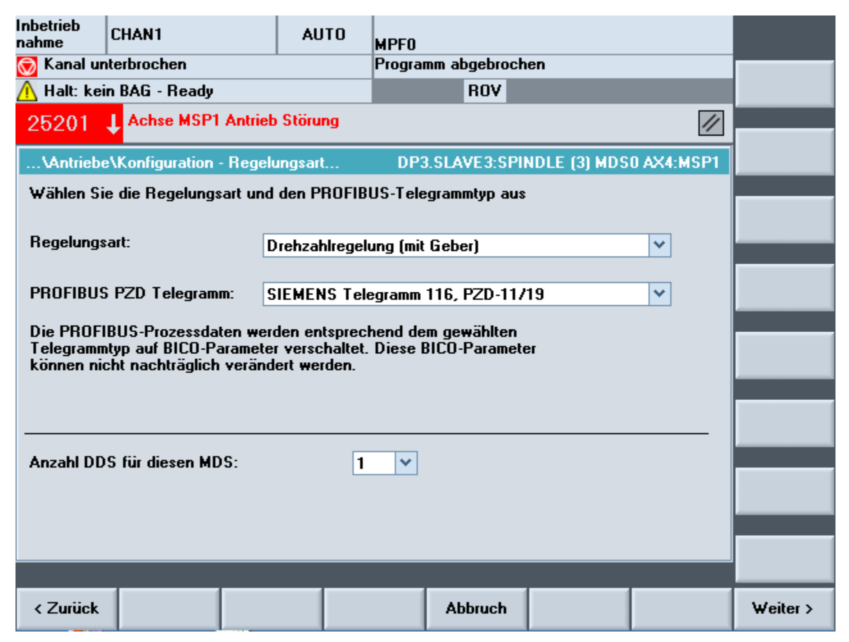


Bild 6-14 Antriebe\Konfiguration - Regelungsart

\- ODER -

10. Wählen Sie die Einstellung mit Telegramm 136 (Momentenvorsteuerung), um für die internen Antriebsgrößen zusätzliche Prozessdaten (PZD) zu übertragen und in Systemvariablen abzulegen.

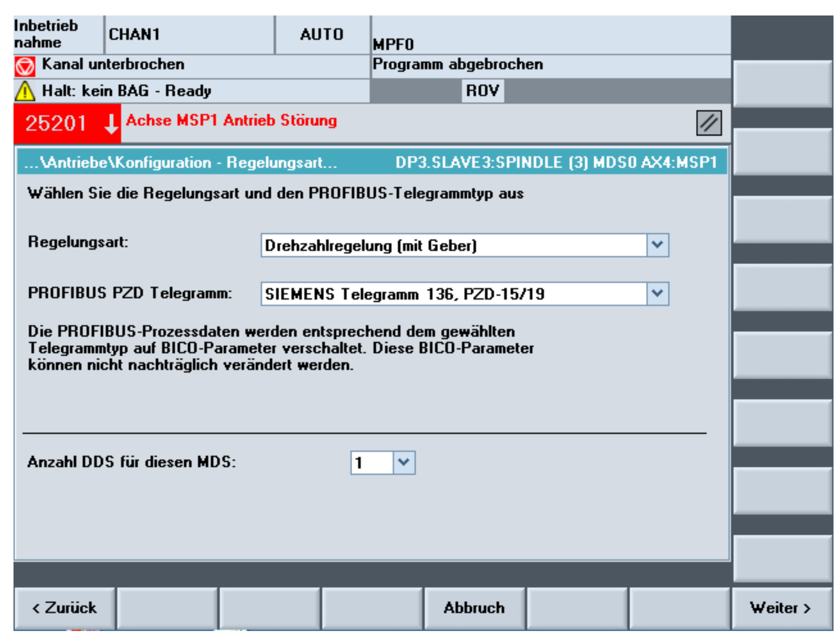


Bild 6-15 Antriebe\Konfiguration - Regelungsart

11. Wählen Sie "Weiter >", um die Auswahl zu speichern. Dieser Vorgang kann einige Minuten dauern.

12. Wählen Sie "Weiter >", um die BICO-Konfiguration zu übernehmen.

    Danach wird folgende Zusammenfassung angezeigt, um alle Daten zur Spindel zu überprüfen:

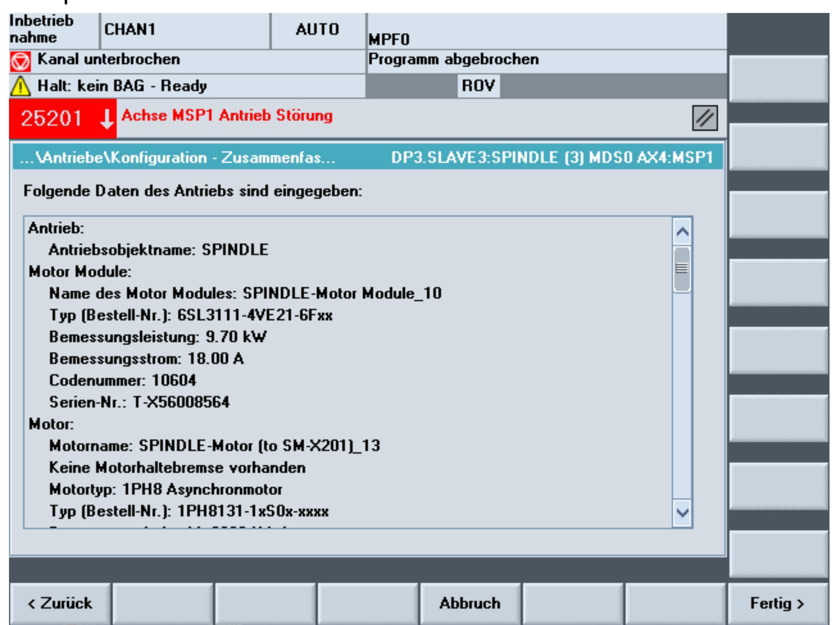


Bild 6-16 Antriebe\Konfiguration - Zusammenfassung

13. Wählen Sie "Fertig >", um die Konfiguration der Spindel abzuschließen.

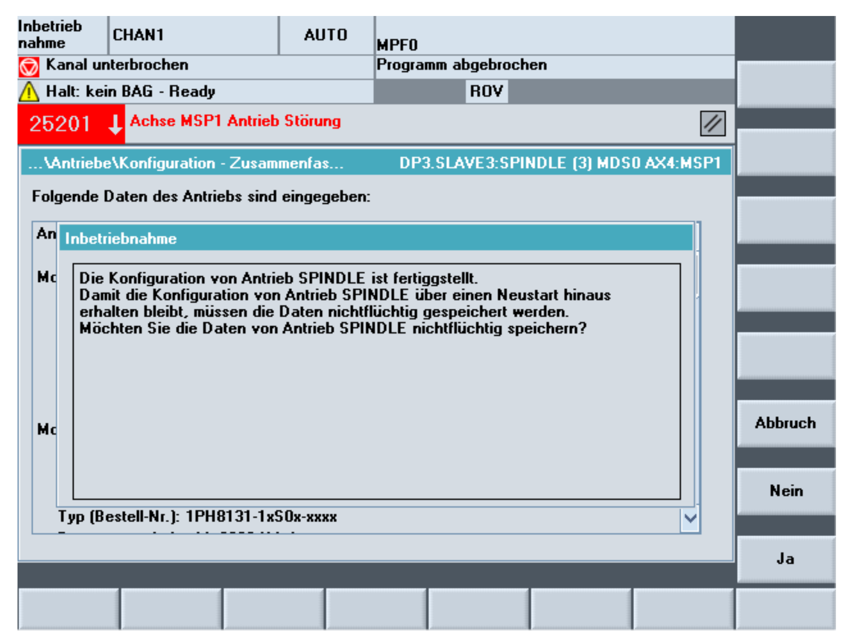


Bild 6-17 Inbetriebnahme - Abfrage

14. Wählen Sie "Ja", um die Konfiguration zu speichern.

Ergebnis

Die Konfiguration der Spindel ist damit abgeschlossen.

6.1.3 Beispiel: So konfigurieren Sie eine Achse mit SMI-Geber

Konfiguration der Achse mit SMI-Geber

Vorgehensweise:

1. Wählen Sie zuerst mit dem Softkey "Antrieb +" oder "Antrieb -" die Achse aus.
2. Um die Achse zu konfigurieren, wählen Sie "Ändern".

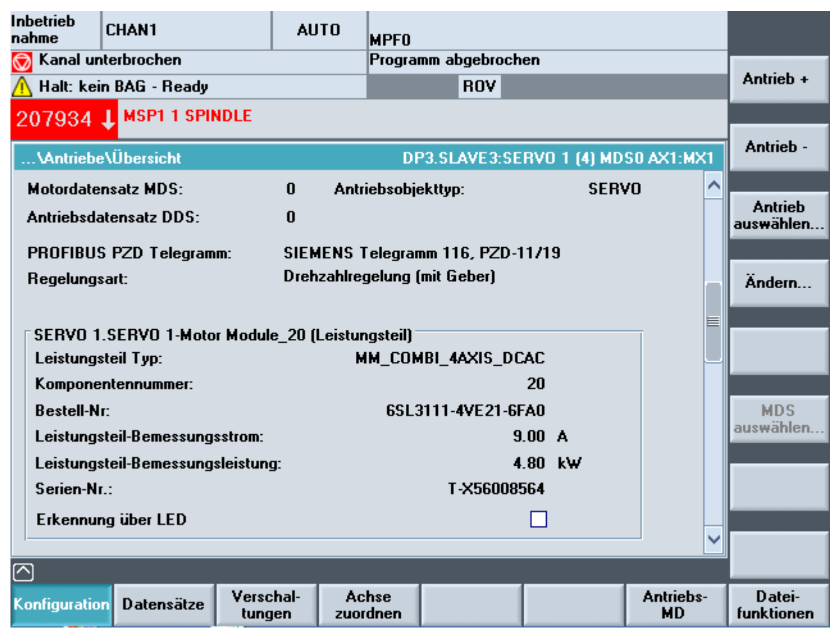


Bild 6-18 Antriebe\Übersicht

Es werden folgende Achsdaten angezeigt:

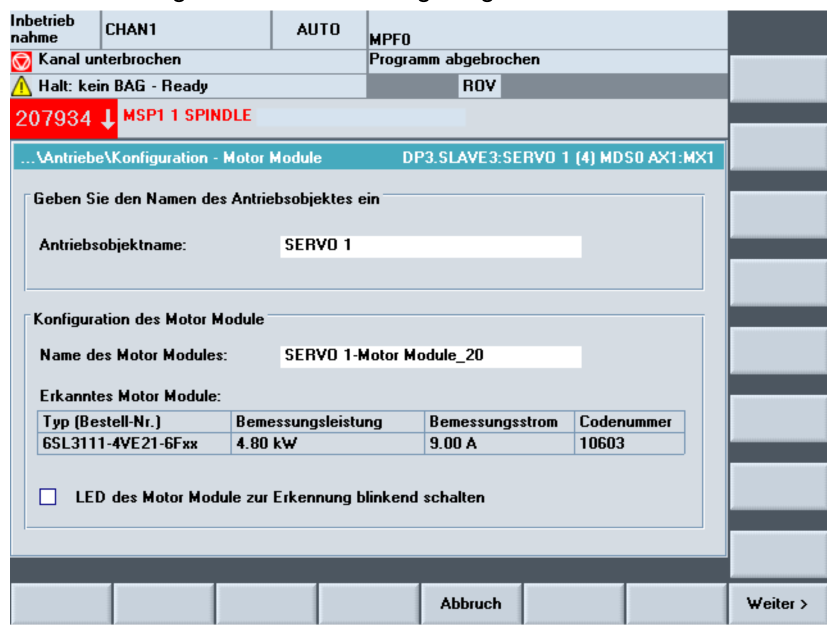


Bild 6-19 Antriebe\Konfiguration - Motor Module

3. Wählen Sie Wählen Sie "Weiter >", um den zugehörigen Motor zu konfigurieren: Der Motortyp wurde bereits korrekt bei der automatischen Ermittlung der Topologie erkannt.

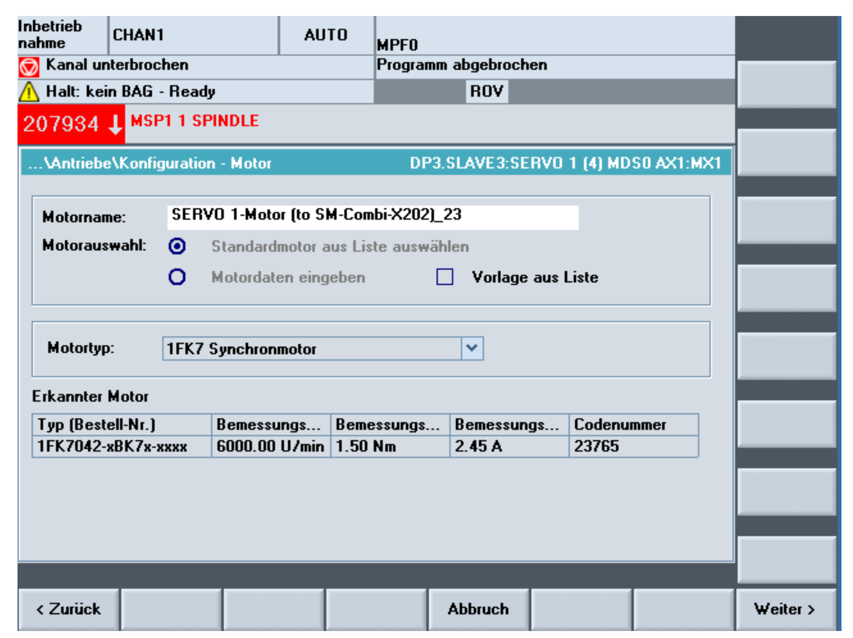


Bild 6-20 Antriebe\Konfiguration - Motor

4. Wählen Sie "Weiter >", um die Motorhaltebremse zu konfigurieren.

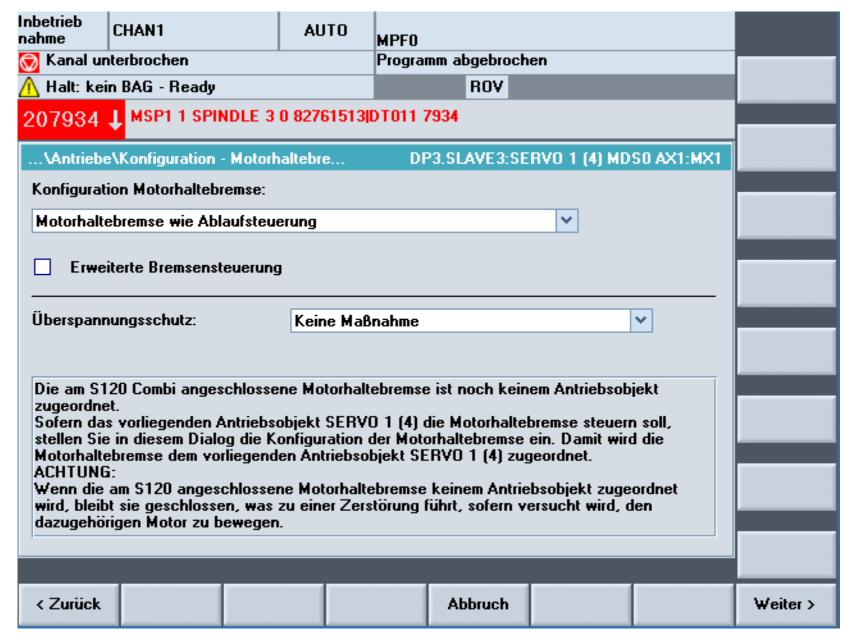


Bild 6-21 Antriebe\Konfiguration - Motorhaltebremse

Beachten Sie den Hinweis zum Überspannungsschutz. Zur Auswahl stehen:

– Keine Maßnahme
– VPM (Voltage Protection Mode)

5. Wählen Sie "Weiter >", um die Geber zuzuordnen: die beiden Geber (am Motormess-System und am direkten Mess-System) wurden bereits korrekt bei der automatischen Ermittlung der Topologie zugeordnet.

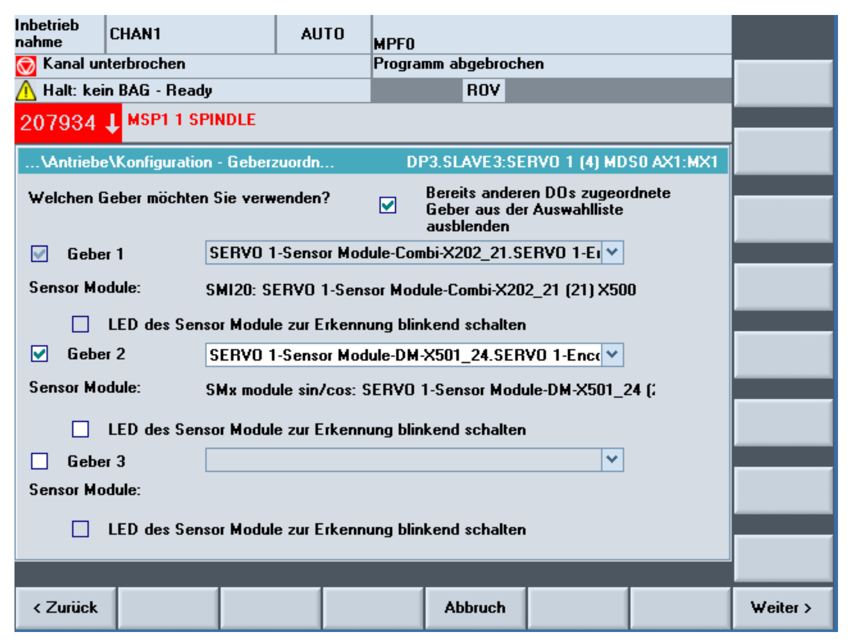


Bild 6-22 Antriebe\Konfiguration - Geberzuordnung

6. Wählen Sie "Weiter >", um den Motorgeber zu konfigurieren.

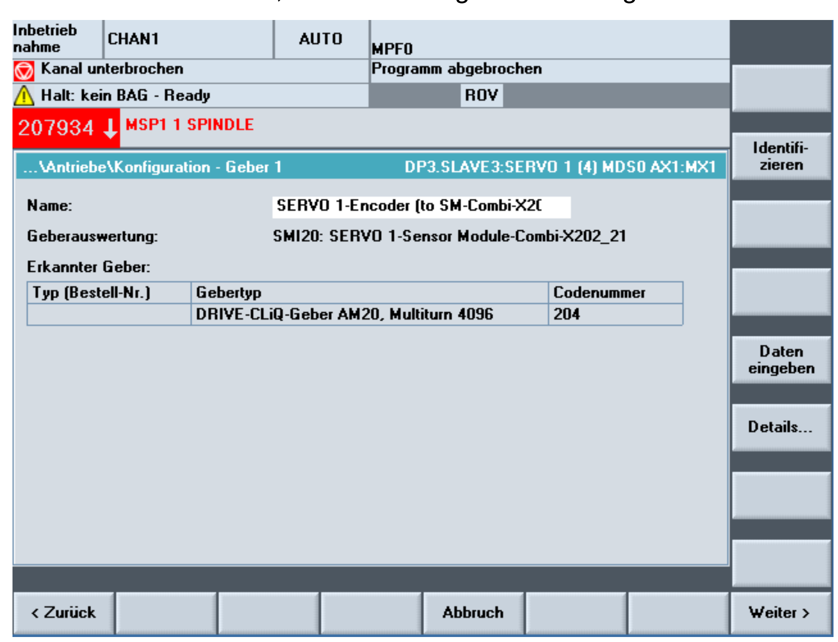


Bild 6-23 Antriebe\Konfiguration - Geber 1

7. Wählen Sie "Daten eingeben", um die Eigenschaften des Gebers auf "Lageistwert invertieren" anzupassen.

8. Wenn Sie diese Einstellung mit "OK" bestätigen, wird der Geber als "Benutzerdefiniert" angezeigt:

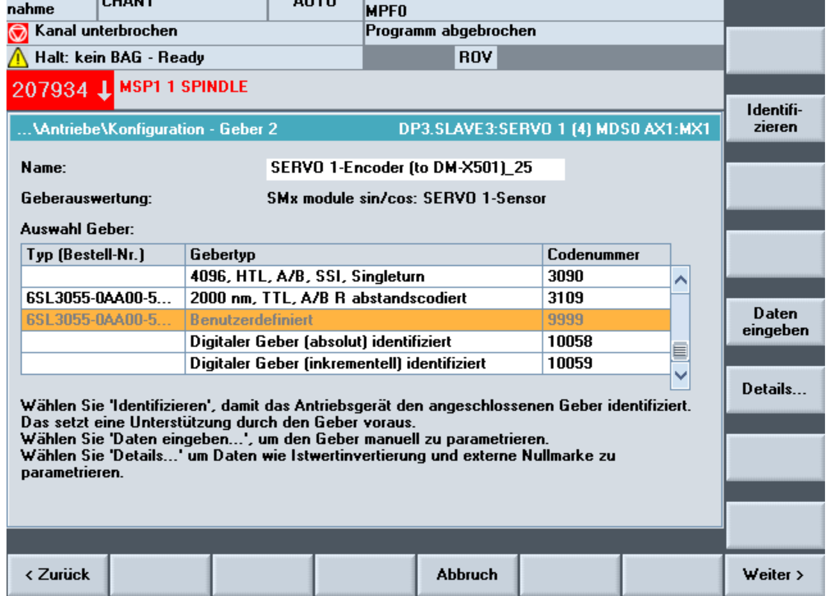


Bild 6-24 Antriebe\Konfiguration - Geber 2

9. Wählen Sie "Weiter >", um die Voreinstellung mit Telegramm 116 (Drehzahlvorsteuerung) zu übernehmen. Damit werden für die internen Antriebsgrößen zusätzliche Prozessdaten (PZD) übertragen und in Systemvariablen abgelegt.

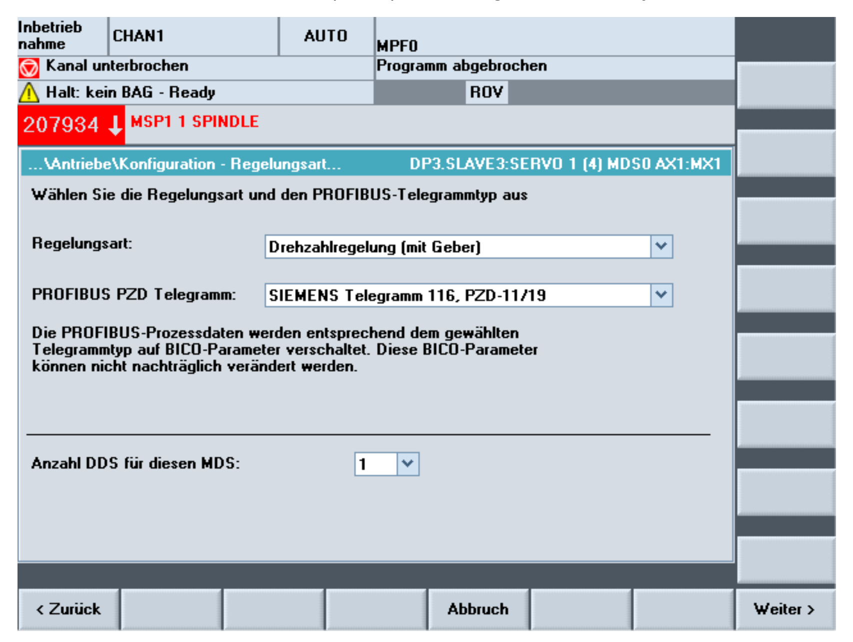


Bild 6-25 Antriebe\Konfiguration - Regelungsart

- ODER -

10. Wählen Sie die Einstellung mit Telegramm 136 (Momentenvorsteuerung), um für die internen Antriebsgrößen zusätzliche Prozessdaten (PZD) zu übertragen und in Systemvariablen abzulegen.

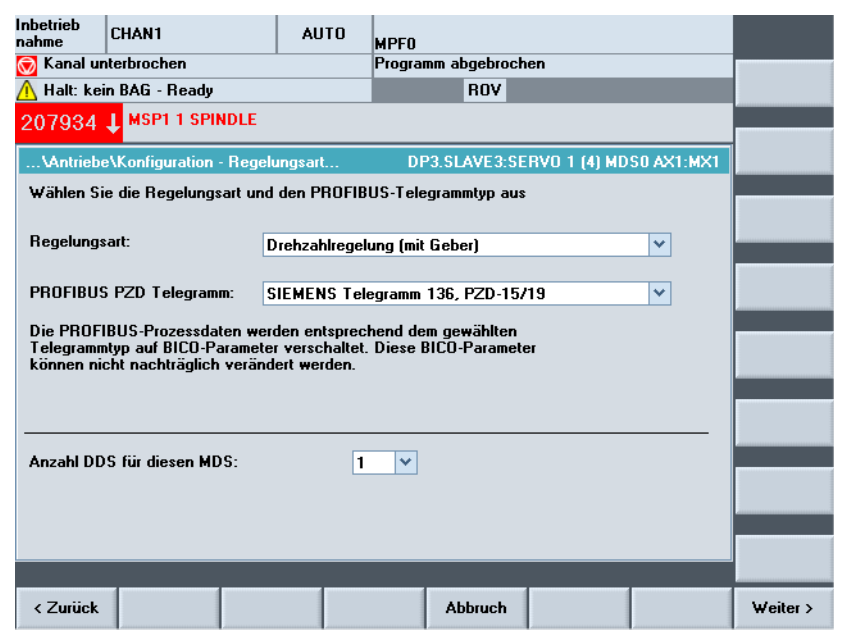


Bild 6-26 Antriebe\Konfiguration - Regelungsart

11. Wählen Sie "Weiter >", um die BICO-Konfiguration zu übernehmen.

    Danach wird folgende Zusammenfassung angezeigt, um alle Daten zur Achse zu überprüfen:

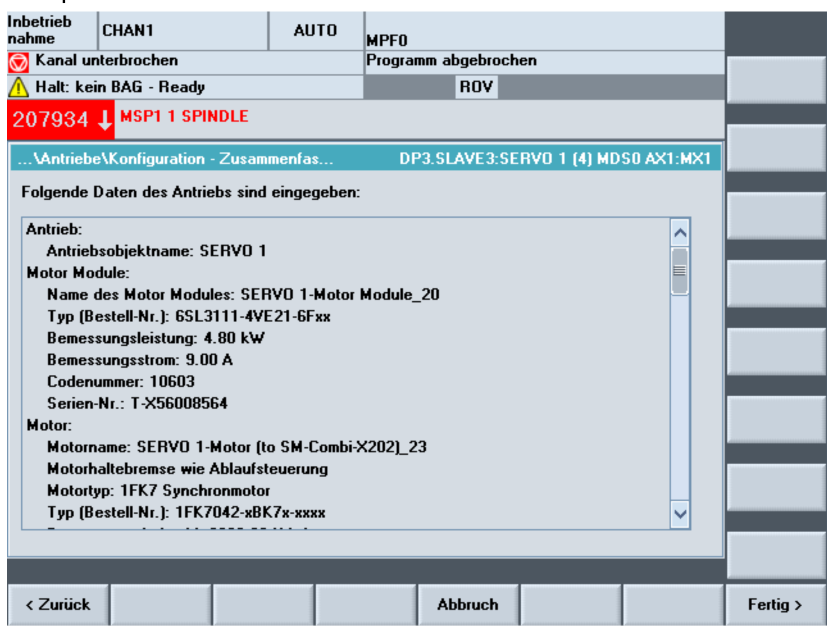


Bild 6-27 Antriebe\Konfiguration - Zusammenfassung

12. Wählen Sie "Fertig >", um die Konfiguration der Achse abzuschließen.
13. Wählen Sie "Ja", um die Konfiguration zu speichern. Dieser Vorgang kann einige Minuten dauern.

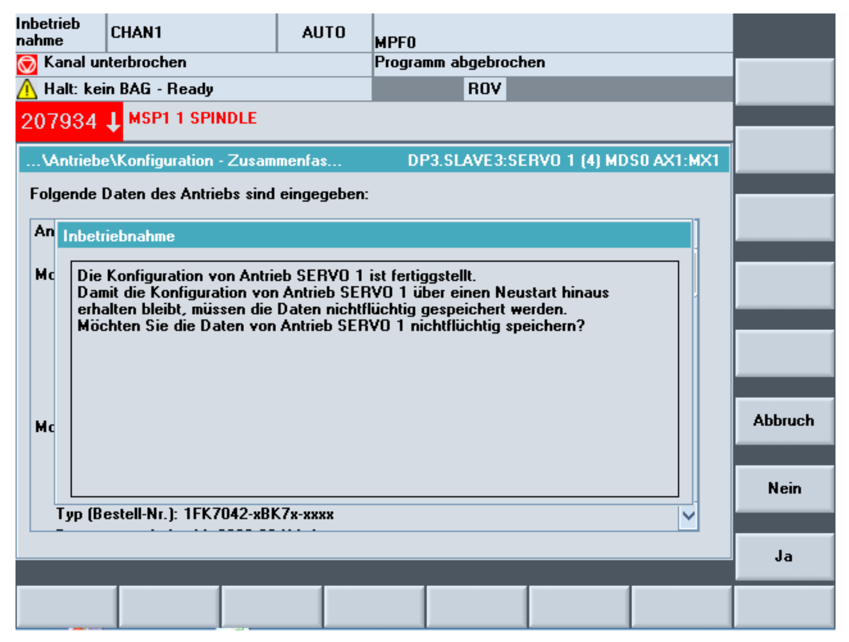


Bild 6-28 Inbetriebnahme - Abfrage

Ergebnis

Die Konfiguration der Achse ist damit abgeschlossen.

6.1.4 Beispiel: Konfiguration mit SINAMICS S120 Booksize

Übersicht

Die Inbetriebnahme-Software für SINAMICS S120 wird kostenlos auf der Toolbox CD zur Verfügung gestellt.

Bis zur vollständigen Verfügbarkeit der SINAMICS S120 Inbetriebnahme-Funktionalität in der Bedienoberfläche, erfolgt die Inbetriebnahme des Antriebs über die Inbetriebnahme-Software für SINAMICS S120. Die Verbindung des PG/PC erfolgt über die frontseitige Ethernet-Schnittstelle der SINUMERIK 828D.

Konfiguration des Antriebs

Für die Konfiguration des Antriebs wird das Beispiel mit 4 Achsen und SINAMICS S120 Booksize aus Kapitel "Systemübersicht (Seite 11)" gewählt. Die DRIVE-CLiQ-Verbindungen sind wie in der nachfolgenden Abbildung angeschlossen:

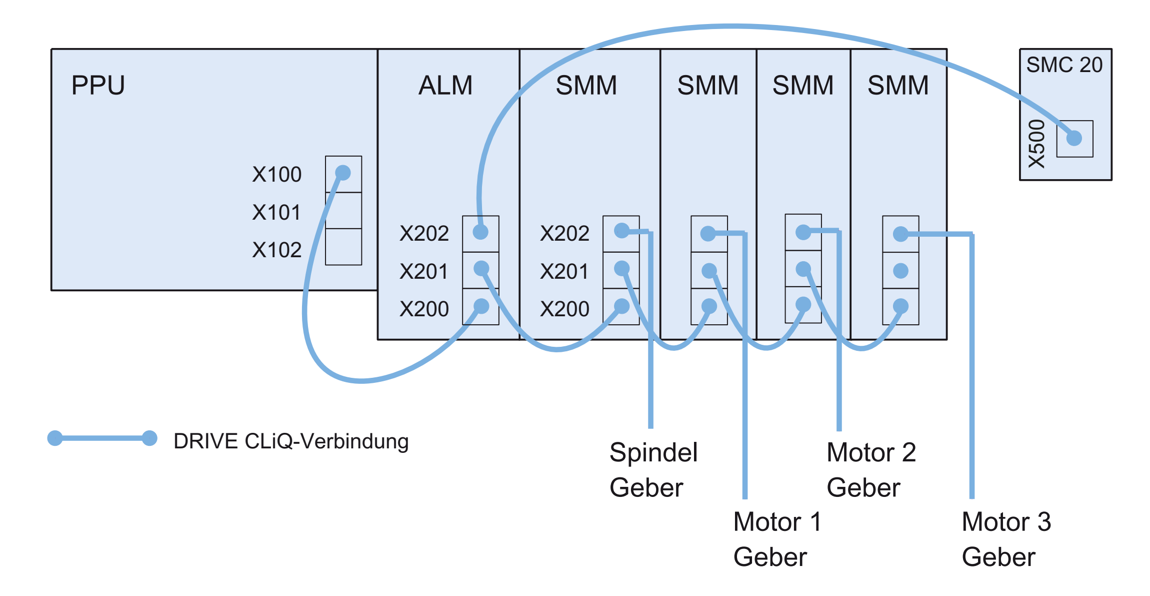


Bild 6-29 DRIVE-CLiQ-Verbindungen

Für die Konfiguration von 4 Achsen gilt:

Die Reihenfolge der DRIVE-CLiQ-Verbindungen entspricht der Reihenfolge der SINAMICS Antriebsobjekt-Nummer (= Voreinstellung). Nur wenn diese Einstellung nicht zu der Reihenfolge im Antriebsverbund passt, sind Anpassungen erforderlich.

Index	Achsen	SINAMICS Antriebsobjekt	
		Nummer	Name
		1	CU_I_3.3:1
		2	SLM_3.3:2
4	MSP1	3	SERVO_3.3:3
1	MX1	4	SERVO_3.3:4
2	MY1	5	SERVO_3.3:5
3	MZ1	6	SERVO_3.3:6

Ablauf

Der Ablauf ist in folgende Schritte aufgeteilt:

- Verbindung zur Steuerung herstellen.
- Schritt 1: Antriebssystem konfigurieren.
- Schritt 2: Einspeisung konfigurieren.
- Schritt 3: Antriebsobjekt konfigurieren.

- Schritt 4: Geber zuordnen.
- Schritt 5: Achsen zuordnen.
- Zum Abschluss: Daten sichern.

Die Schritte sind detailliert in den nachfolgenden Kapiteln beschrieben.

Siehe auch

Topologieregeln für S120 Booksize (Seite 168)

6.1.5 Beispiel: So konfigurieren Sie das Antriebssystem

Ausgangszustand

Bevor Sie beginnen:

- PG/PC mit der Steuerung verbinden: siehe Kapitel Beispiel: So kommunizieren Sie über den NCU Connection Wizard mit der Steuerung (Seite 29)
- Der Hochlauf der Steuerung wird mit "Siemens default data" durchgeführt.
- Anzeige im Bedienbereich "Inbetriebnahme" an der Steuerung:

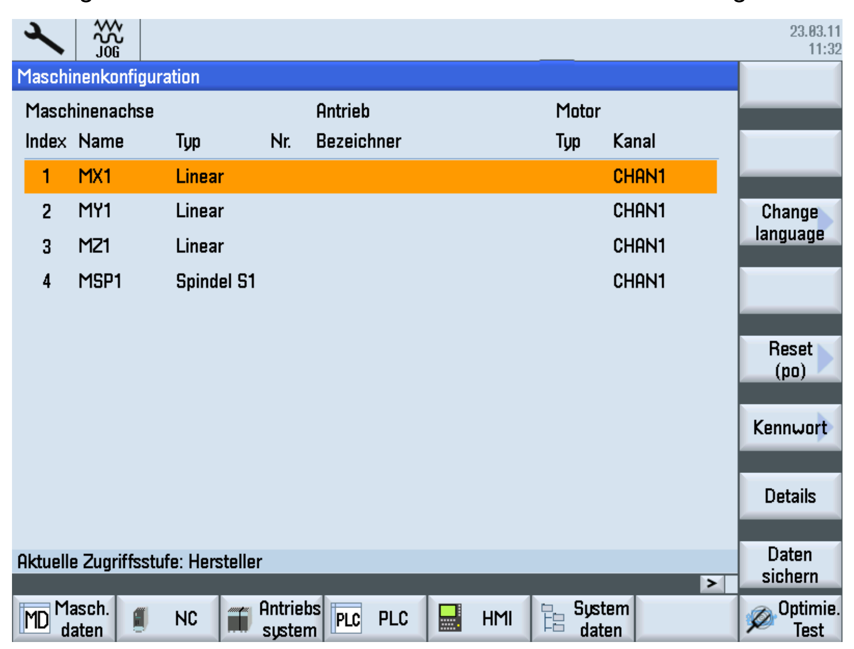


Bild 6-30 Maschinenkonfiguration nach Hochlauf

Vorgehensweise

Um den Antrieb zu konfigurieren:

1. Starten Sie die Inbetriebnahme-Software für SINAMICS S120 auf PG/PC:



2. Wählen Sie den Bedienbereich "Inbetriebnahme".
3. Setzen Sie das Kennwort auf Zugriffsstufe "Hersteller". Danach erhalten Sie folgende Anzeige:

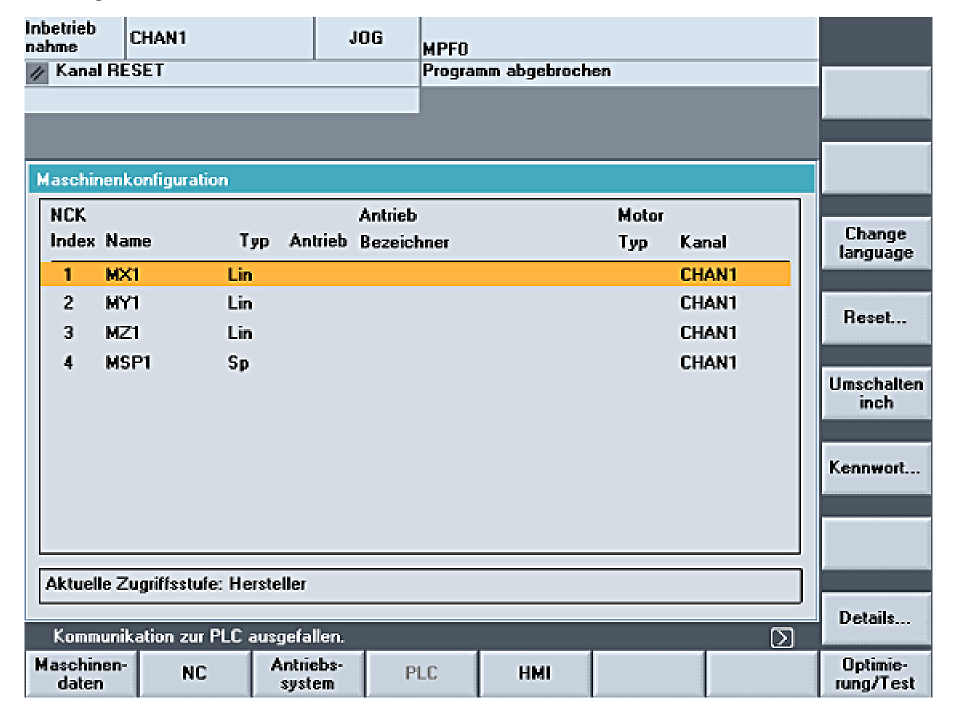


Bild 6-31 Maschinenkonfiguration: Start der Inbetriebnahme

Hinweis

Firmware aktualisieren

Wenn ein Antrieb angeschlossen ist, wird die Aktualisierung der Firmware automatisch gestartet.

- Während des Ladens werden Sie mit einer Zustandsanzeige über den Fortschritt informiert.
- Während die Firmware geladen wird, blinkt am jeweiligen Modul die LED "RDY" abwechselnd rot - grün. Wenn das Modul fertig ist, leuchtet diese LED konstant grün. Die LED "DC LINK" leuchtet konstant "orange".

---

4. Wenn das Laden der Firmware erfolgreich beendet worden ist, bestätigen Sie diese Meldung mit "OK":

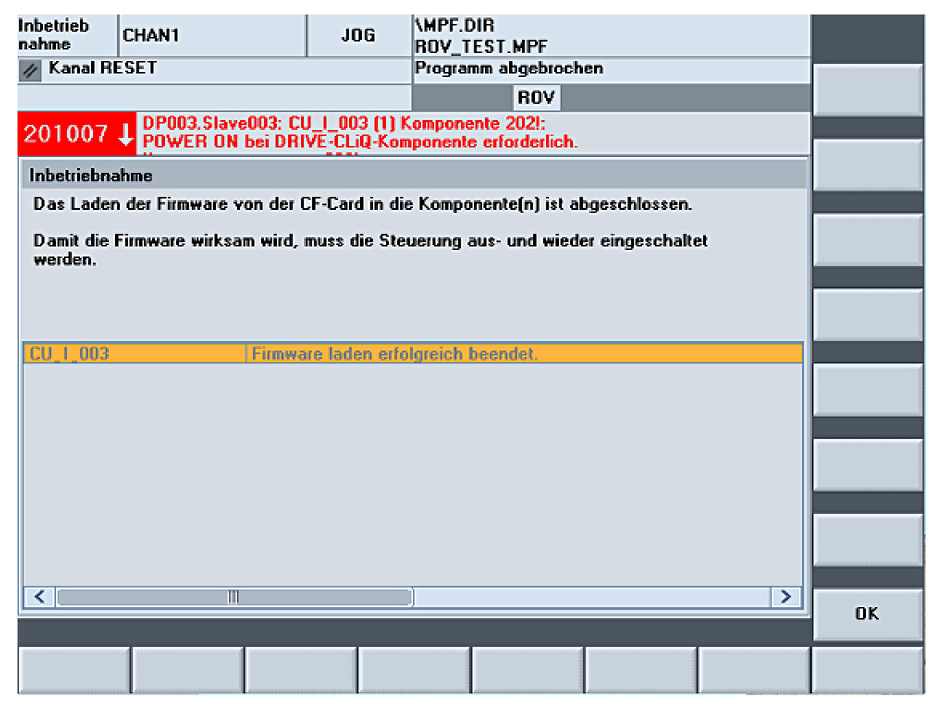


Bild 6-32 Inbetriebnahme: Firmware laden

5. Führen Sie - wie in der Meldung gefordert - ein POWER ON für den Antrieb durch: Schalten Sie komplett aus und danach wieder ein.

   Die folgende Auswahl ist abhängig von der Konfiguration des Zielsystems:

   – **ohne** NX Erweiterungsbaugruppe: Softkey "Antriebsgerät"

   – **mit** NX Erweiterungsbaugruppe: Softkey "Antriebssystem"

   Diese Beispiel-Konfiguration ist eine Topologie ohne NX-Erweiterungsbaugruppe.

6. Wählen Sie in dafür den Softkey "Antriebsgerät" (vertikale Leiste) und setzen Sie die Konfiguration mit "OK" fort.

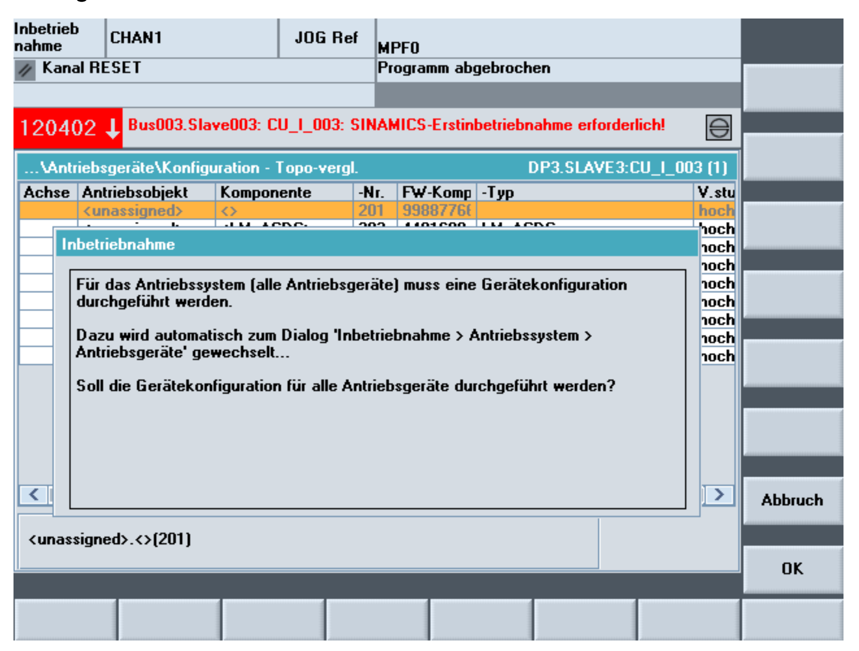


Bild 6-33 Antriebsgeräte\Konfiguration - Topologievergleich

7. Danach erhalten Sie die Meldung, dass der Vorgang der Gerätekonfiguration mehrere Minuten dauern kann.

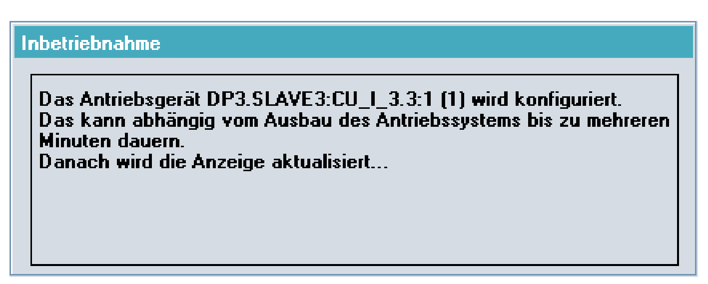


Bild 6-34 Meldung zur Konfiguration

8. Danach erhalten Sie folgende Anzeige:

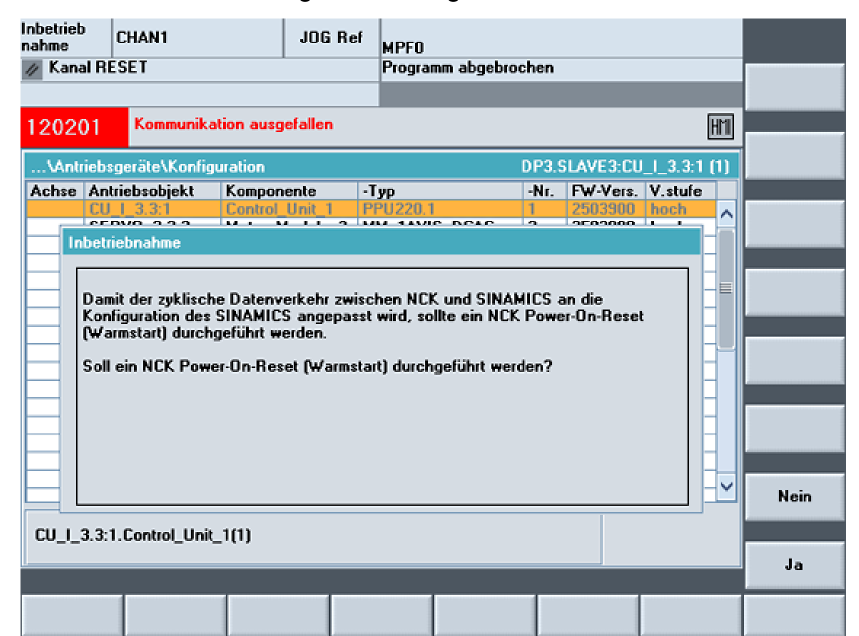


Bild 6-35 Antriebsgeräte\Konfiguration: NCK Power-On-Reset

9. Bestätigen Sie mit "Ja". Das System führt einen Warmstart durch. Dieser Vorgang kann einige Minuten dauern. Nach dem Warmstart erhalten Sie folgende Auswahl:

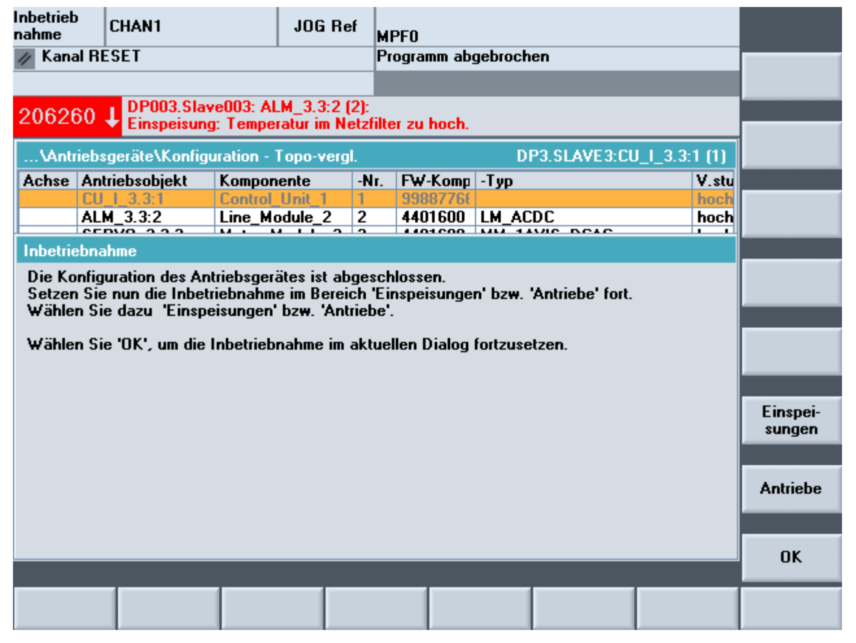


Bild 6-36 Antriebsgeräte\Konfiguration: Inbetriebnahme fortsetzen

10. Um mit der Inbetriebnahme fortzufahren, drücken Sie den Softkey "Antriebe".

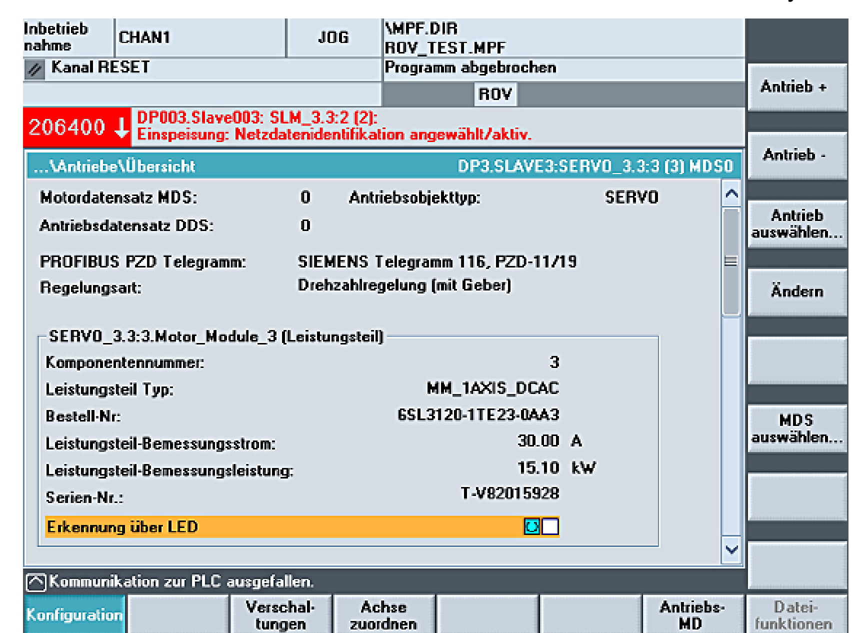


Bild 6-37 Antriebe\Übersicht

11. Option: Um die einzelnen Motor Module zu identifizieren, wählen Sie "Erkennung über LED": Die LED "RDY" blinkt abwechselnd rot-orange.

    Mit den vertikalen Softkeys "Antrieb +" und "Antrieb -" wählen Sie das nächste Modul an.

Ergebnis

Schritt 1 für die Inbetriebnahme der Antriebe ist damit abgeschlossen.

6.1.6 Beispiel: So konfigurieren Sie die Einspeisung

Einspeisung konfigurieren

Hinweis

Falls nach dem Einschalten Alarme anstehen, die eine Quittung verlangen, müssen diese zuerst quittiert werden. Danach kann die Inbetriebnahme fortgesetzt werden.

Vorgehensweise:

1. Wählen Sie den horizontalen Softkey "Einspeisung", um folgende Übersicht zu erhalten:

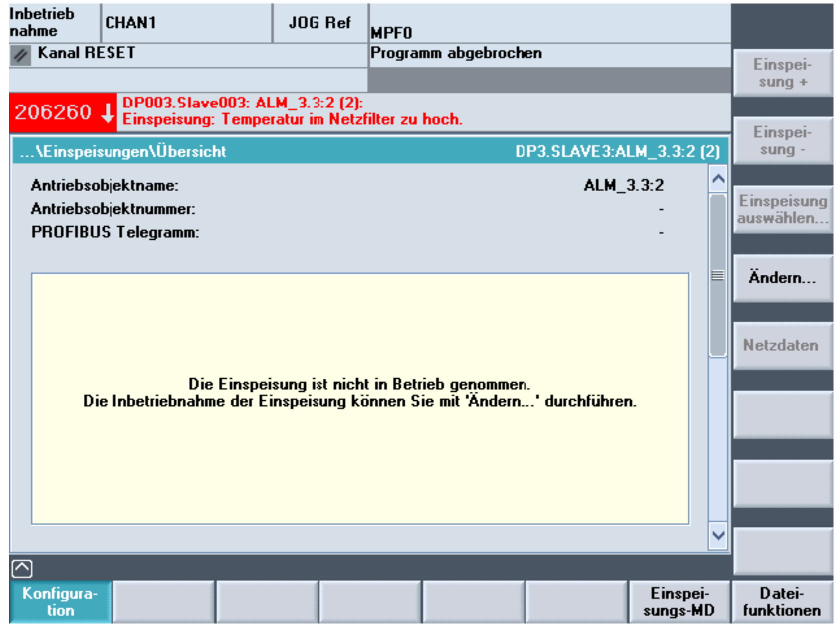


Bild 6-38 Einspeisungen\Übersicht

2. Um die Einspeisung zu konfigurieren, wählen Sie den vertikalen Softkey "Ändern". Im folgenden Dialog kann ein neuer Name für die Einspeisung vergeben werden oder Sie übernehmen die Voreinstellung:

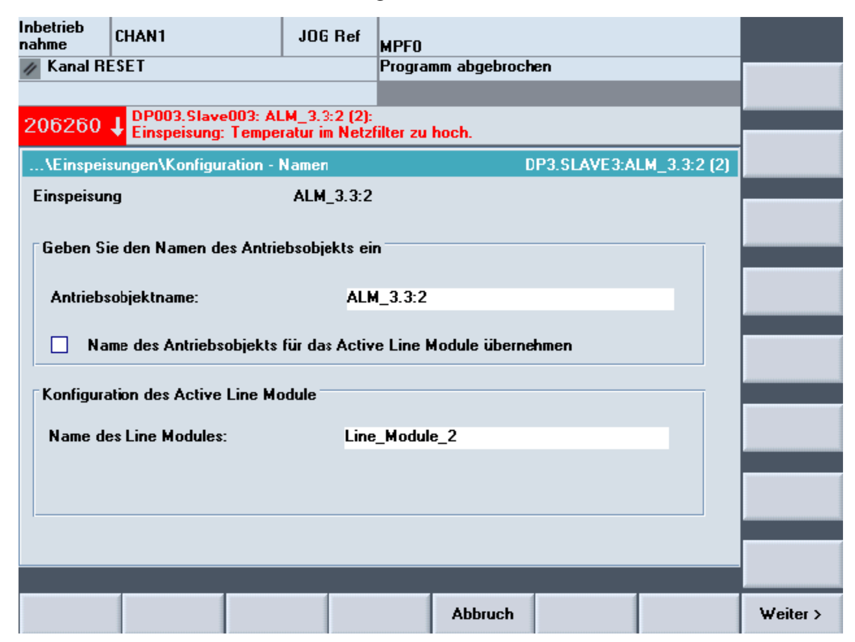


Bild 6-39 Einspeisungen\Konfiguration - Namen

3. Im folgenden Dialog "Konfiguration - Line Module" wird das automatisch erkannte Line Module und der angeschlossene Netzfilter angezeigt. Drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren oder wählen Sie die Option "Bremsmodul extern" an, falls ein Bremsmodul vorhanden ist.

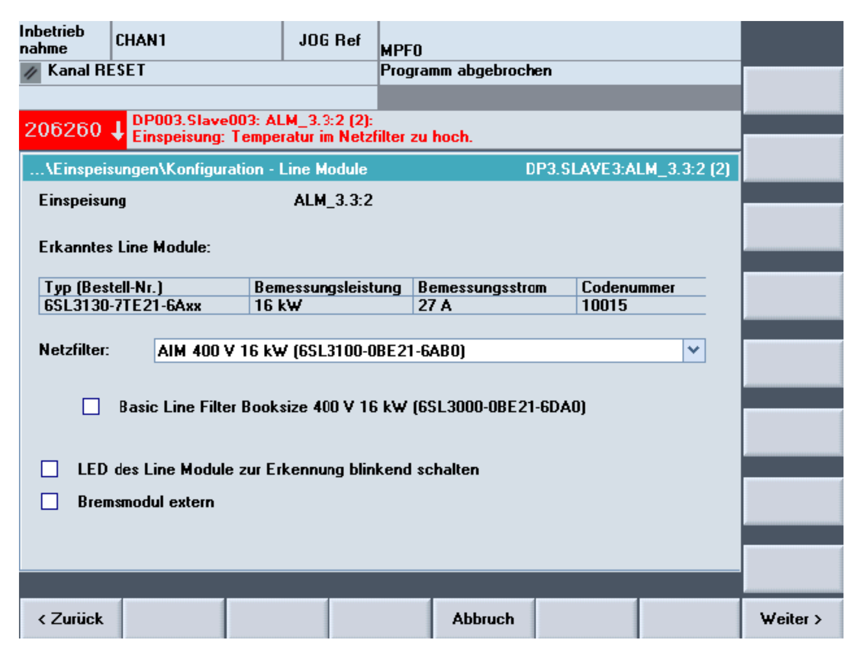


Bild 6-40 Einspeisungen\Konfiguration - Line Module

4. Im folgenden Dialog "Konfiguration - Weitere Daten" wurde die Netz-/Zwischenkreis-identifikation ermittelt und gespeichert. Drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren oder wählen Sie eine Geräteanschluss-Spannung und einen Telegrammtyp aus.

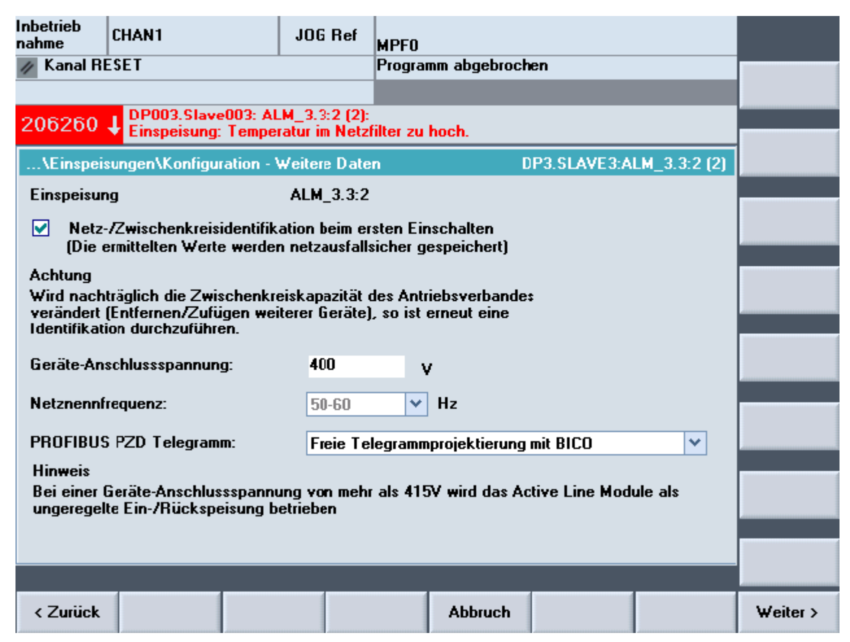


Bild 6-41 Einspeisungen\Konfiguration - Weitere Daten

5. Im folgenden Dialog "Konfiguration - Klemmenverdrahtung" wählen Sie die SINAMICS-interne Netzschützansteuerung oder drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren.

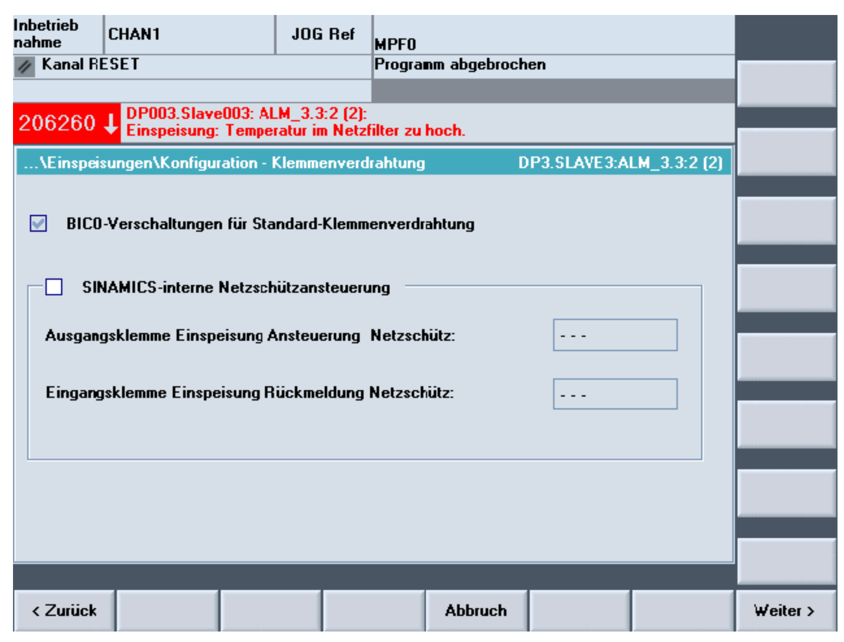


Bild 6-42 Einspeisungen\Konfiguration - Klemmenverdrahtung

6. Im folgenden Dialog "Konfiguration - Zusammenfassung" werden die zur Einspeisung eingegebenen Daten angezeigt. Drücken Sie "Fertig", um die Konfiguration abzuschließen oder "Zurück", wenn Sie weitere Einstellungen vornehmen wollen.

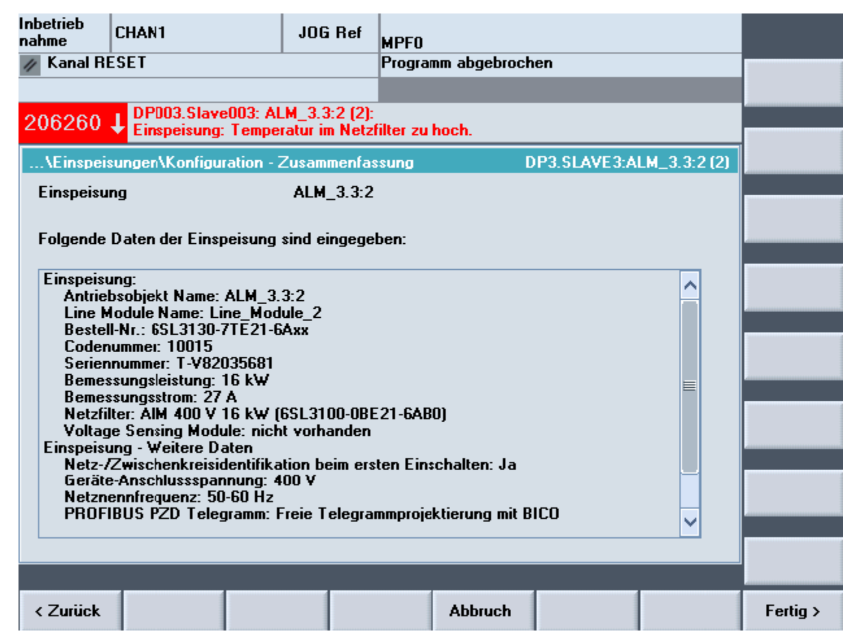


Bild 6-43 Einspeisungen\Konfiguration - Zusammenfassung

7. Bestätigen Sie die Frage zum Speichern der Daten mit "Ja". Dieser Vorgang kann einige Minuten dauern.

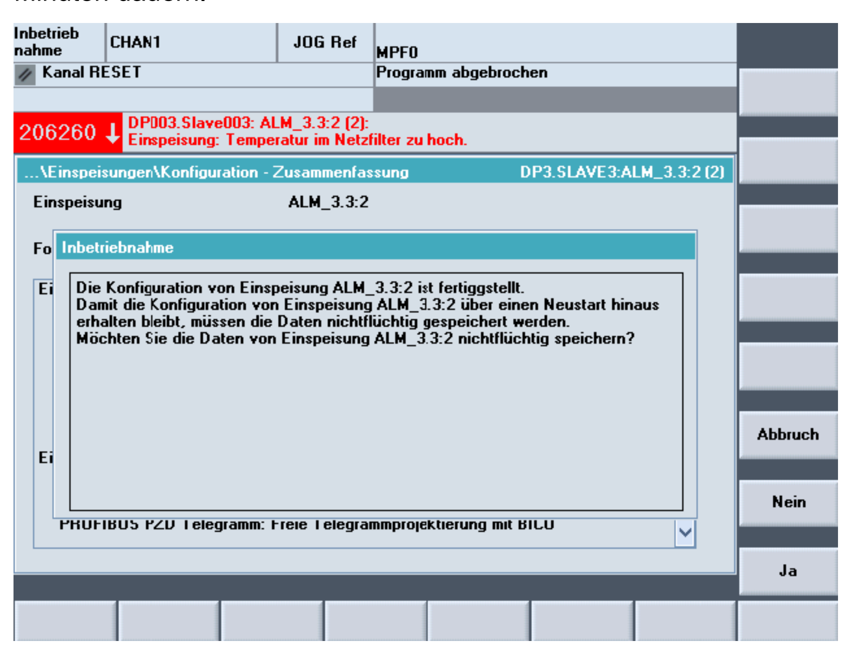


Bild 6-44 Einspeisungen\Konfiguration - Zusammenfassung: Abfrage

8. Im folgenden Dialog "Übersicht" erhalten Sie eine Übersicht über die Konfigurationsdaten des Antriebsobjekts:

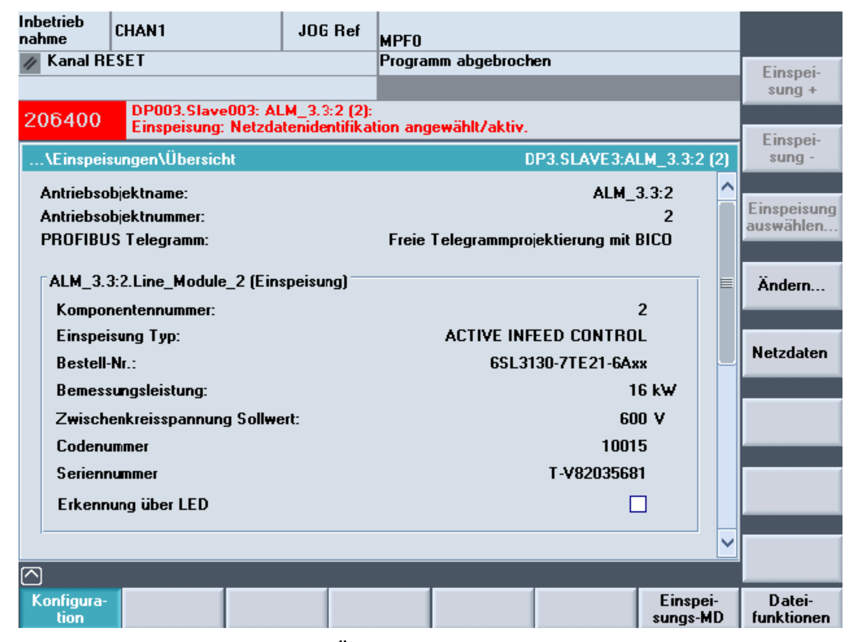


Bild 6-45 Einspeisungen\Übersicht

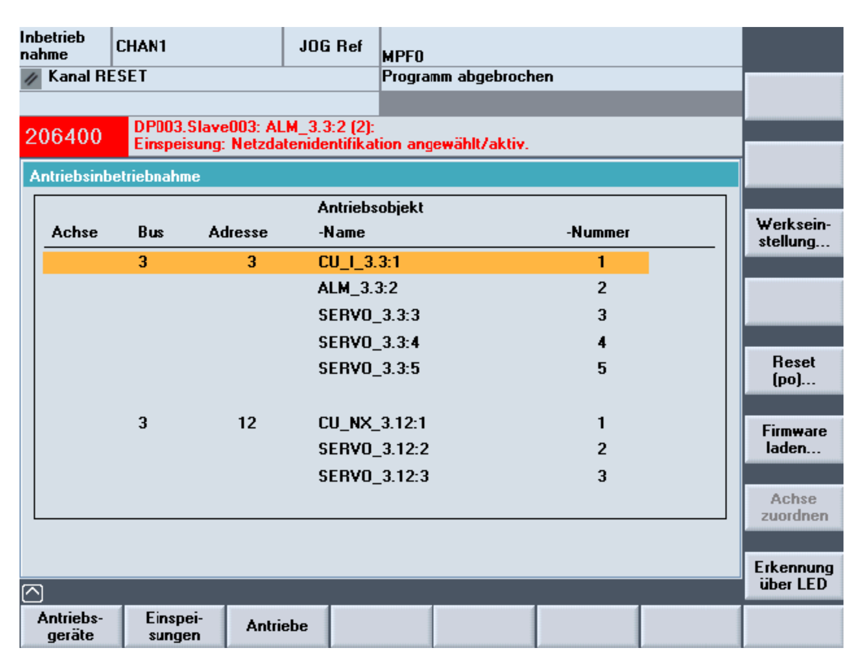
Ergebnis

Schritt 2 für die Inbetriebnahme der Einspeisung ist damit abgeschlossen.

6.1.7 Beispiel: So konfigurieren Sie ein Antriebsobjekt

Ausgangszustand

In der Übersicht "Antriebsinbetriebnahme" werden folgende Antriebsobjekte (DO) angezeigt:



Antriebsinbetriebnahme

Vorgehensweise

Um ein Antriebsobjekt zu konfigurieren:

1. Wählen Sie mit den Cursor-Tasten ein Antriebsobjekt aus und drücken Sie den Softkey "Antriebe". Wenn das Antriebsobjekt noch nicht in Betrieb genommen ist, wählen Sie "Ändern".

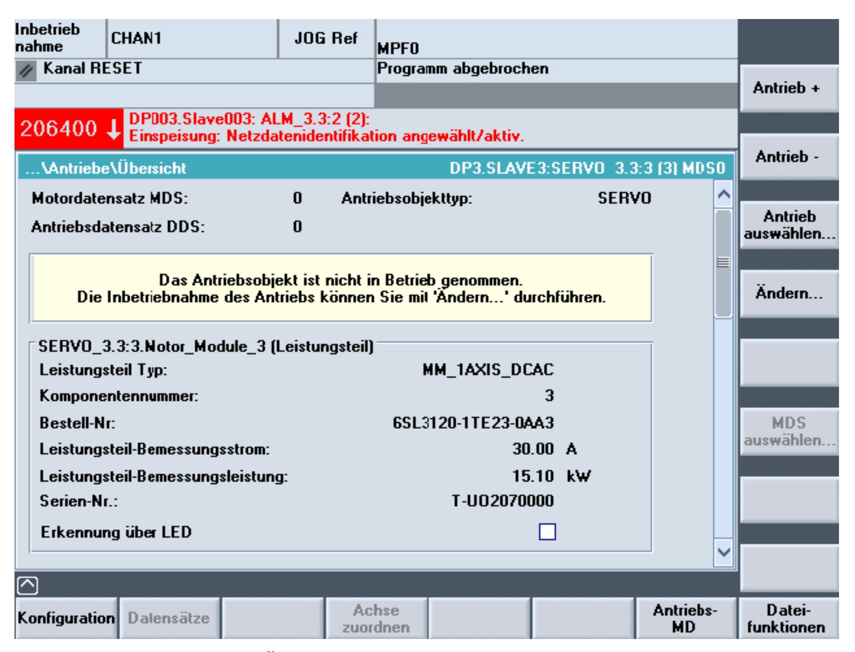


Bild 6-46 Antriebe\Übersicht: Start der Inbetriebnahme

2. Im folgenden Dialog "Konfiguration - Motor Module" kann ein neuer Name für das Motor Module vergeben werden oder Sie übernehmen die Voreinstellung:

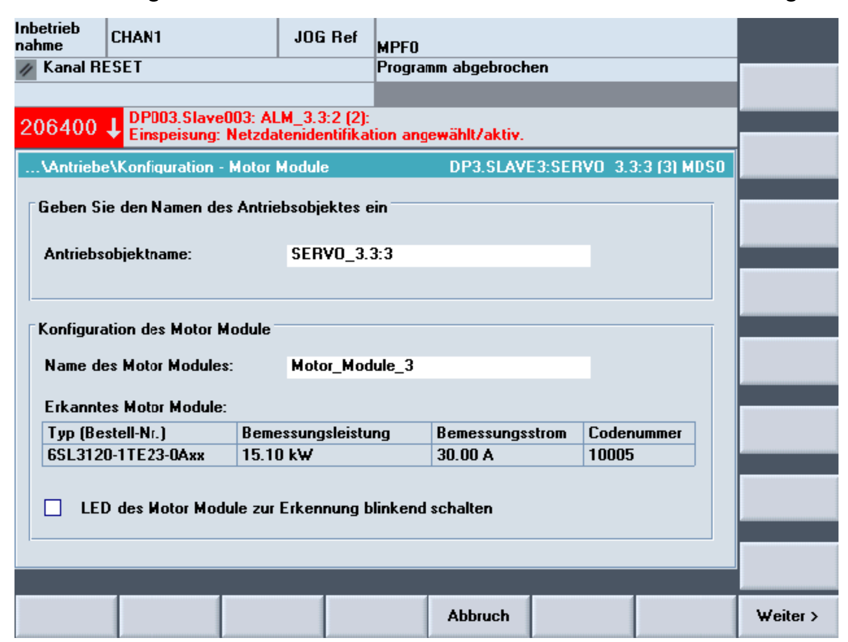


Bild 6-47 Antriebe\Konfiguration - Motor Module

3. Im folgenden Dialog "Konfiguration - Motor" wählen Sie aus der Liste der Motoren den angeschlossenen Motortyp aus. Drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren.

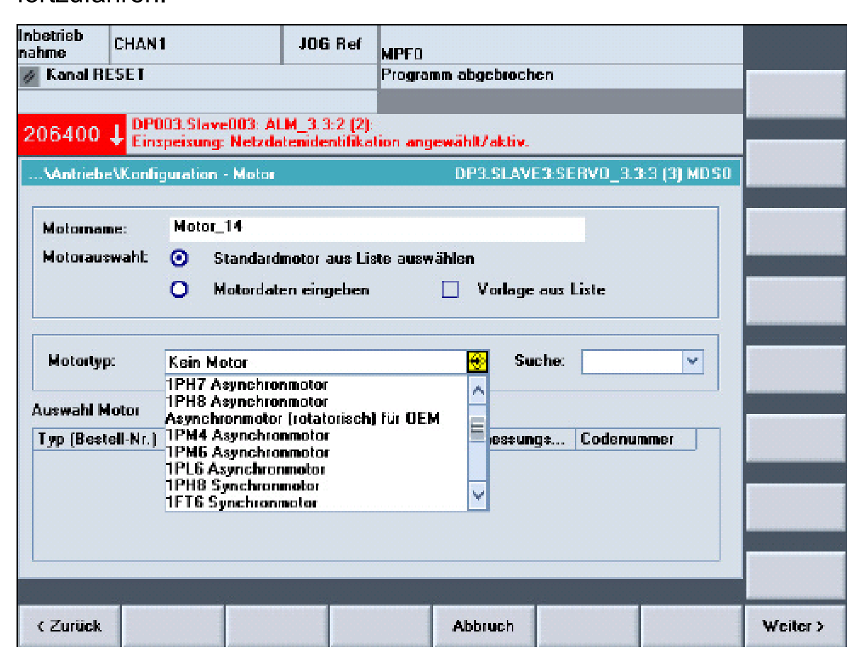


Bild 6-48 Antriebe\Konfiguration - Motor

4. Im folgenden Dialog "Konfiguration - Motorhaltebremse" wählen Sie die Motorhaltebremse aus, falls vorhanden, und selektieren die erweiterte Bremsenansteuerung. Drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren.

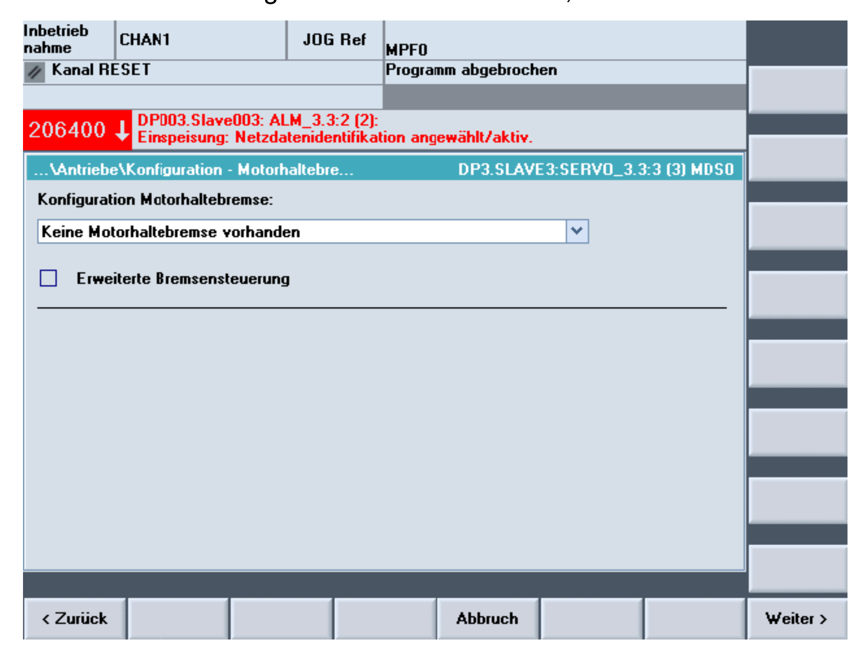


Bild 6-49 Antriebe\Konfiguration - Motorhaltebremse

5. Im folgenden Dialog "Konfiguration - Geberzuordnung" wählen Sie den am Motor angeschlossenen Geber aus der Liste der im Antriebsverbund vorhandenen Geber aus. Drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren.

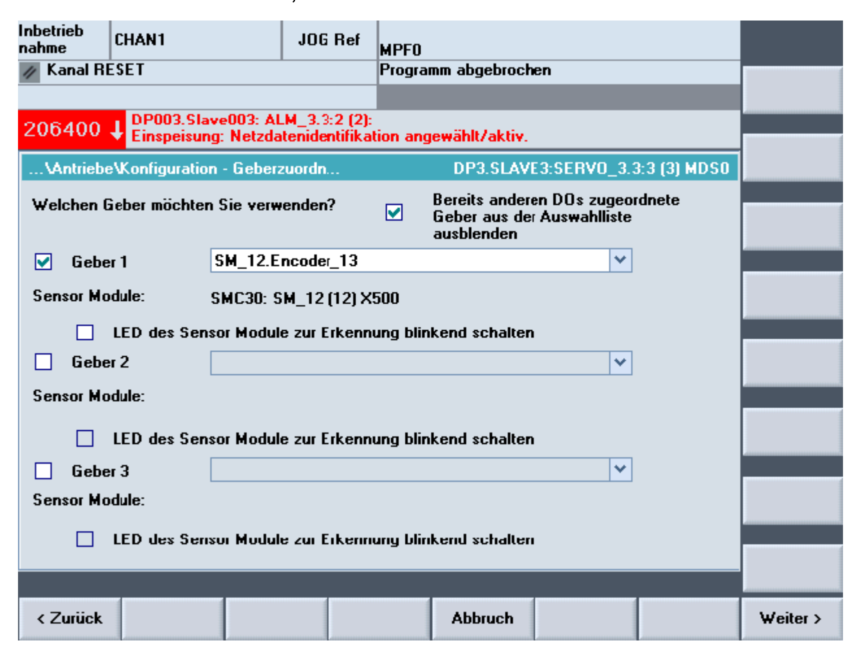


Bild 6-50 Antriebe\Konfiguration - Geberzuordnung

6. Im folgenden Dialog "Konfiguration - Geber 1" wählen Sie unter "Auswahl Motorgeber" den Gebertyp aus.
   - Um einen Geber zu konfigurieren, der nicht in der Auswahlliste angeboten wird, wählen Sie "Daten eingeben".
   - Wählen Sie "Details", um Konfigurationsdaten des Gebers einzugeben.

Drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren.

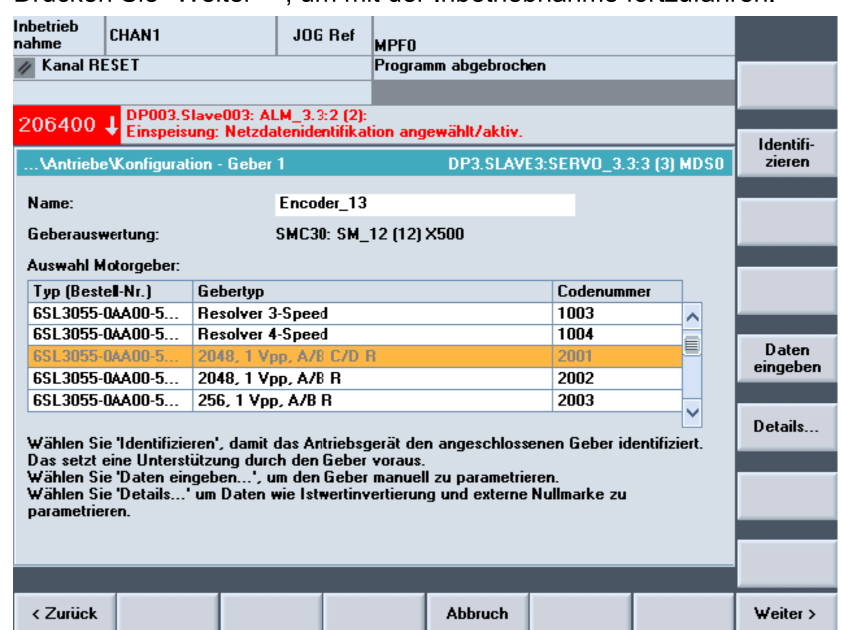


Bild 6-51 Antriebe\Konfiguration - Geber 1

7. Im folgenden Dialog "Konfiguration - Regelungsart" wählen Sie die Regelungsart mit oder ohne Geber sowie das zugehörige PROFIBUS-Telegramm aus. Die Regelungsart und das PROFIBUS-Telegramm sind in der Regel bereits korrekt vorbelegt.

   Außerdem wählen Sie hier die Anzahl der Antriebsdatensätze (DDS) für diesen Motordatensatz (MDS). Die Voreinstellung ist 1 DDS; es sind maximal 8 DDS pro MDS zulässig. Drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren.

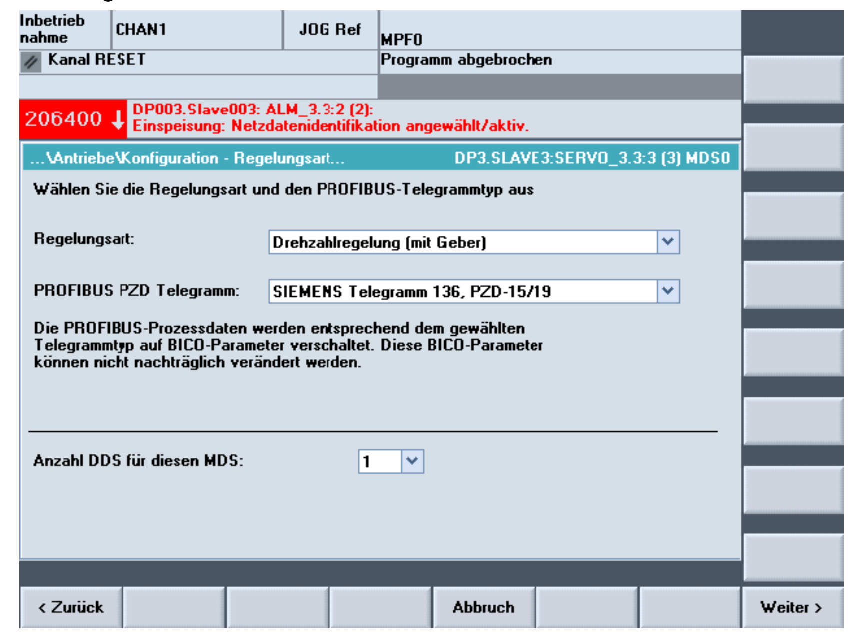


Bild 6-52 Antriebe\Konfiguration - Regelungsart

8. Im folgenden Dialog "Konfiguration - BICO-Verschaltungen" wählen Sie die BICO-Verschaltung für AUS2. Die Voreinstellung ist abhängig vom gewählten Telegramm.

   Drücken Sie "Weiter >", um mit der Inbetriebnahme fortzufahren.

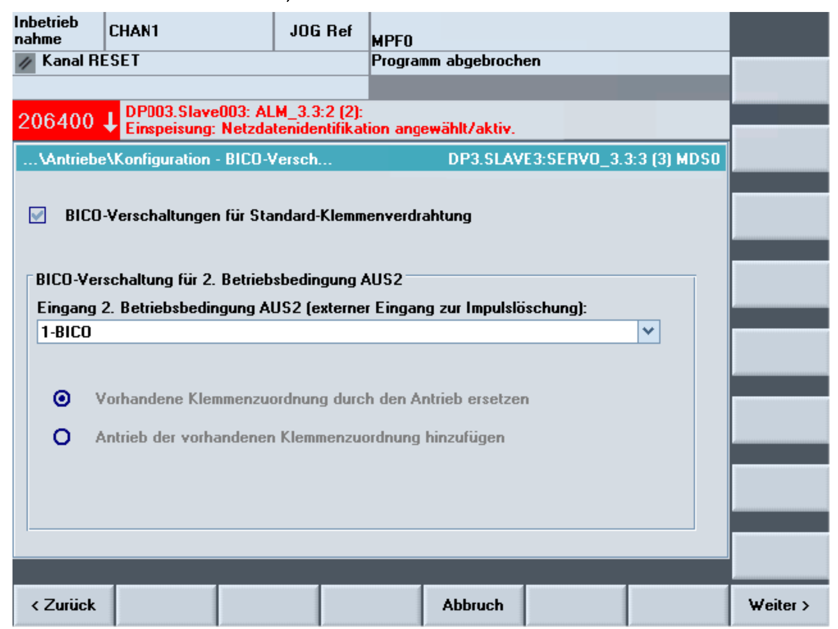


Bild 6-53 Antriebe\Konfiguration - BICO-Verschaltungen

9. Im folgenden Dialog "Konfiguration - Zusammenfassung" werden die eingegebenen Daten des Motor Modules angezeigt. Drücken Sie "Fertig >", um die Inbetriebnahme abschließen.

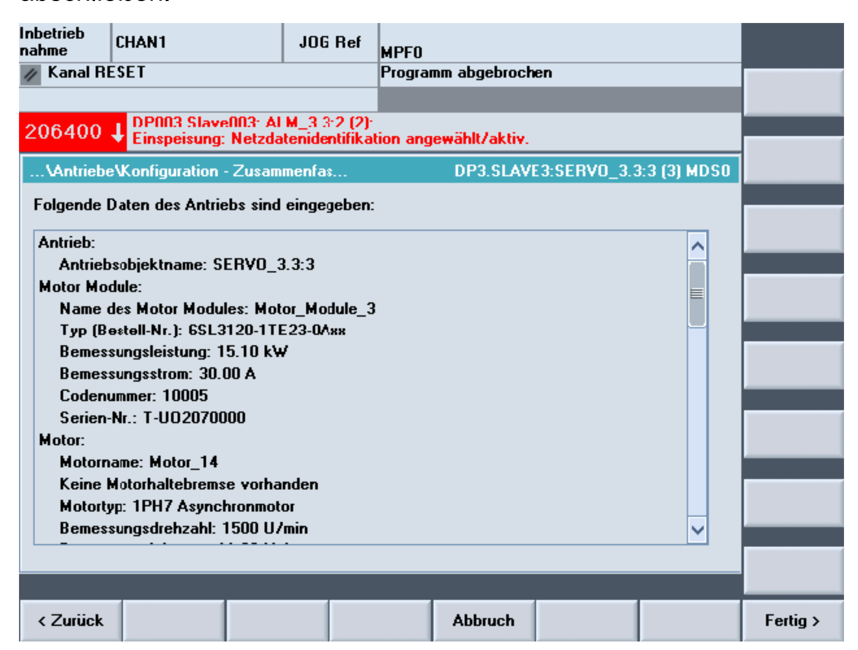


Bild 6-54 Antriebe\Konfiguration - Zusammenfassung

10. In der folgenden Abfrage wählen Sie, ob Sie die Antriebsdaten speichern wollen: Bestätigen Sie mit "Ja>", um die Konfiguration des Antriebs zu sichern.

Bild 6-55 Inbetriebnahme - Abfrage

Ergebnis

Schritt 3 zur Inbetriebnahme eines Antriebsobjektes ist damit abgeschlossen. Wiederholen Sie diese Vorgehensweise für alle Antriebsobjekte.

6.1.8 Beispiel: So konfigurieren Sie den externen Geber

Anschluss eines direkten Mess-Systems

Für die Spindel ist zusätzlich ein direktes Mess-System angeschlossen (siehe Kapitel Beispiel: Konfiguration mit SINAMICS S120 Booksize (Seite 119)). Der folgende Abschnitt beschreibt die Konfiguration.

Vorgehensweise

Um die Konfiguration des Antriebs zu ändern:

1. Sie haben den vertikalen Softkey "Ändern" gedrückt.

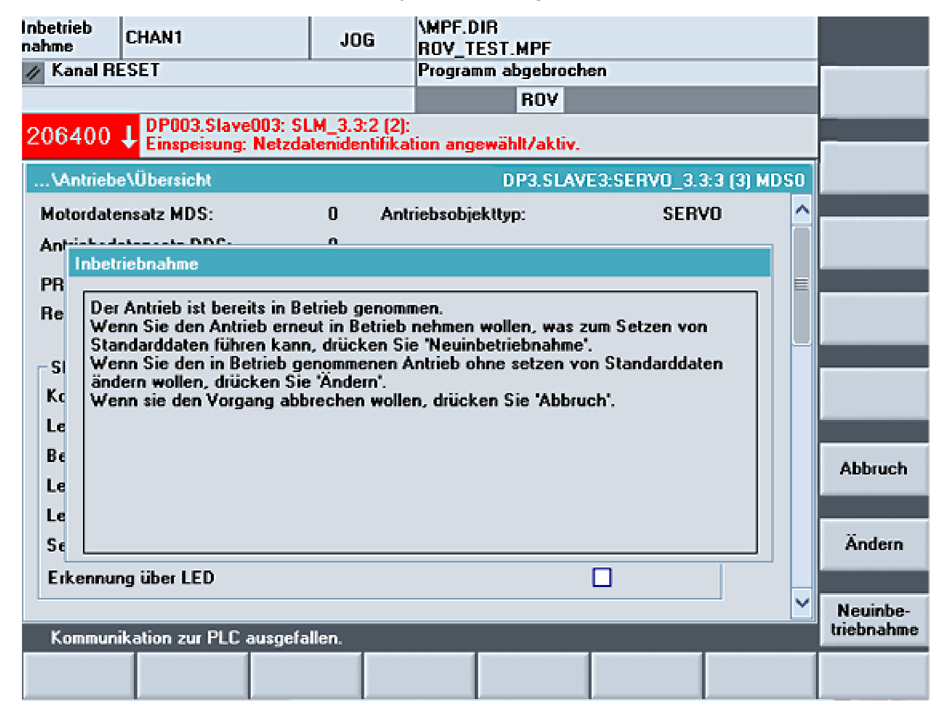


Bild 6-56 Antrieb ändern

2. Bestätigen Sie die Rückfrage mit dem Softkey "Ändern".

   Der Dialog "Konfiguration - Motor" wird geöffnet und das Antriebsobjekt 3 = Spindel ist ausgewählt. Drücken Sie "Weiter >".

3. Der nächste Dialog informiert Sie darüber, welcher Motor dem Antrieb zugeordnet ist.

   Drücken Sie "Weiter >".

4. Der nächste Dialog informiert Sie über die Motorzuordnung. Drücken Sie "Weiter >".

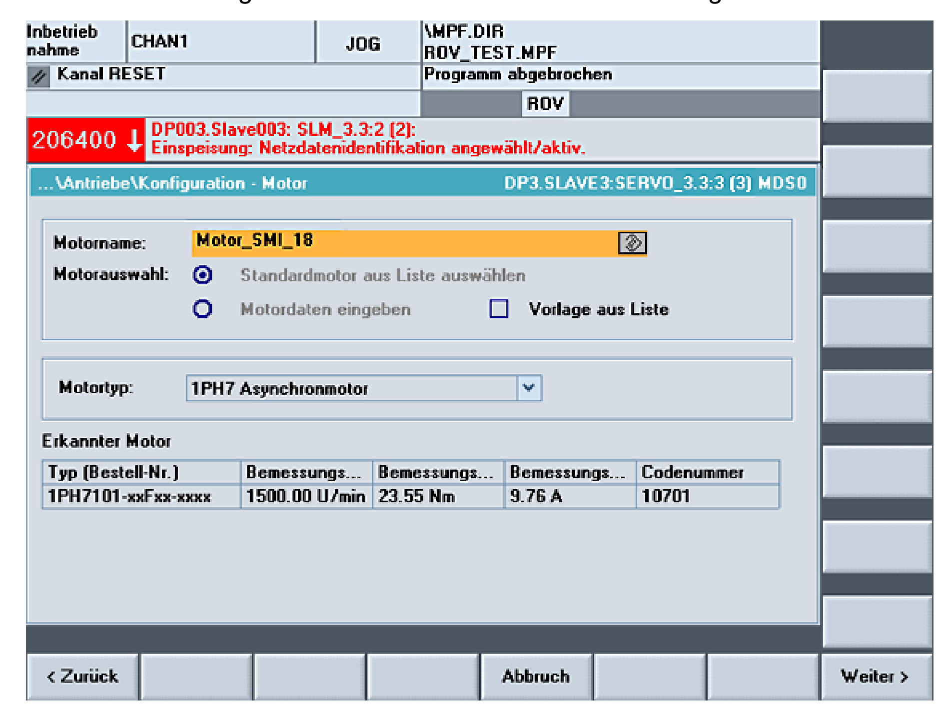


Bild 6-57 Motor konfigurieren

5. Dieser Dialog informiert Sie über die genauen Daten des erkannten Motors.

   Drücken Sie "Weiter >".

6. Der nächste Dialog informiert Sie über die Konfiguration der Motorbremse.

   Drücken Sie "Weiter >".

7. Der nächste Dialog informiert Sie darüber, welcher Geber diesem Antriebsobjekt (= Spindel) bereits zugeordnet ist:

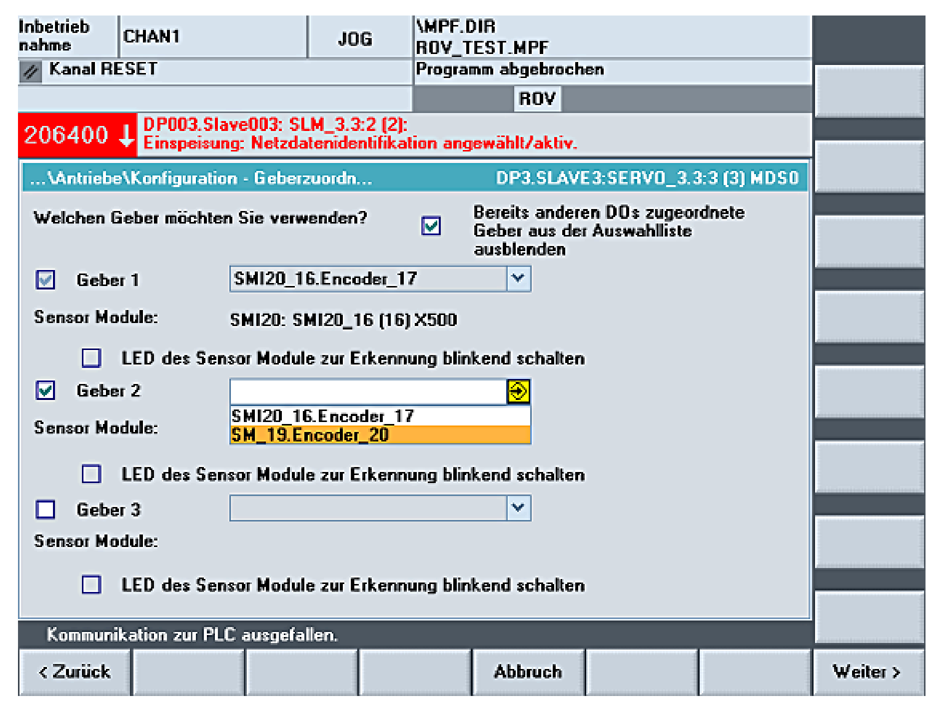


Bild 6-58 Geber zuordnen

8. Aktivieren Sie die Option "Geber 2".
9. Wählen Sie in der Auswahlliste den Geber " SM_19.Encoder_20".
10. Bestätigen Sie mit der Taste <INPUT>.
11. Drücken Sie "Weiter >".
12. Bestätigen Sie die Rückfrage mit "OK".

    Dies kann einige Minuten in Anspruch nehmen.
13. Während die Daten gespeichert werden, erhalten Sie folgende Zustandsanzeige:

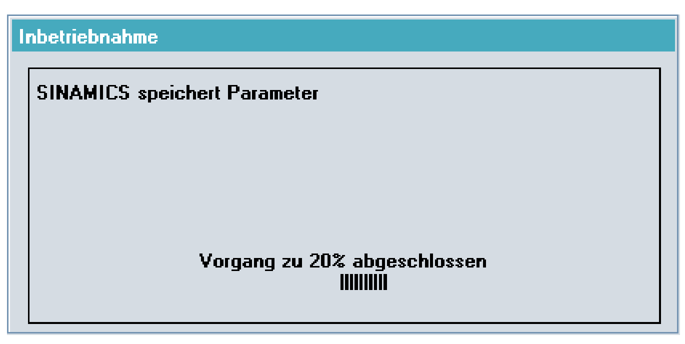


Bild 6-59 Geber 2: Daten speichern

14. Der nächste Dialog informiert Sie über die Konfiguration der Regelungsart.
15. Drücken Sie "Weiter >".

16. Der nächste Dialog informiert Sie über die BICO Verbindung.
17. Drücken Sie "Weiter >".
18. Zum Abschluss erhalten Sie eine Zusammenfassung mit den kompletten Daten angezeigt.
19. Wenn Sie "Fertig >" drücken, erhalten Sie folgende Information:

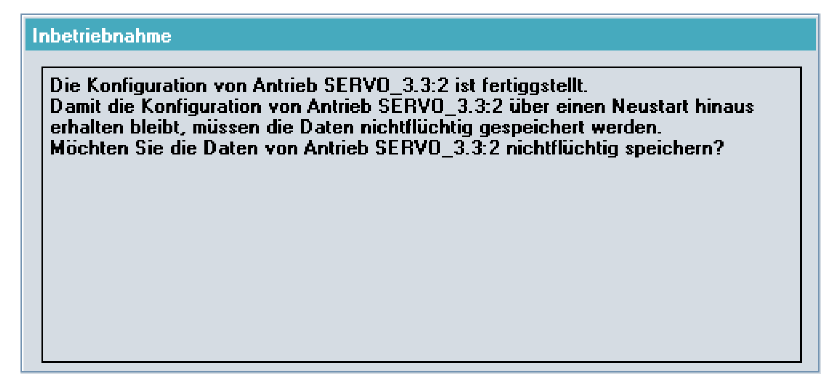


Bild 6-60 Speichern bestätigen

20. Bestätigen Sie das Speichern der Konfigurationsdaten mit "Ja".

    Das Speichern kann einige Minuten dauern.

Ergebnis

Schritt 4 zur Inbetriebnahme des direkten Mess-Systems ist damit abgeschlossen.

6.1.9 Antriebskomponente nach der Erst-Inbetriebnahme tauschen

Komponententausch

Das Tauschen von Komponenten nach der Erst-Inbetriebnahme, z. B. um ein ALM mit höherer Leistung 16 kW → 50 kW einzusetzen, muss so erfolgen, dass die Konfigurationsdaten neu eingelesen werden und wieder aktuell sind.

Hinweis

Firmware-Update

Das ordnungsgemäße Firmware-Update der projektierten Antriebskomponenten ist nur dann gegeben, wenn diese im ausgeschalteten Zustand gesteckt wurden. Das nachträgliche Stecken von Antriebskomponenten darf **nur** im ausgeschalteten Zustand erfolgen.

Wird eine Antriebskomponente nach der Erst-Inbetriebnahme durch eine Komponente mit unterschiedlicher Bestellnummer z. B. ein leistungsstärkeres Modul ersetzt, beachten Sie folgenden Ablauf:

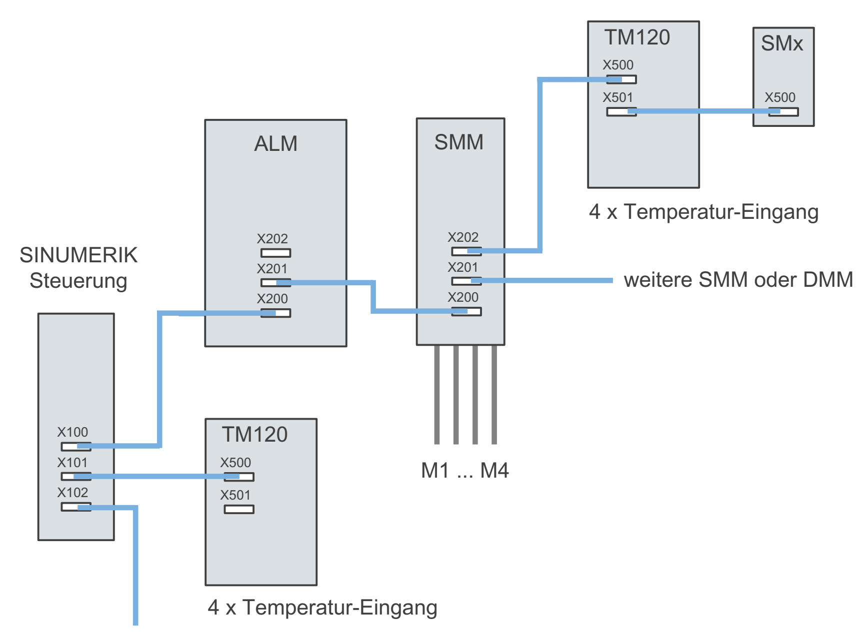
Vorgehensweise:

1. Wählen Sie "Antriebsgerät" → "Topologie" → "Ändern".
2. Selektieren Sie das Modul und wählen Sie "Antriebsobjekt deaktivieren".
3. Tauschen Sie das Modul im Schrank aus.
4. Selektieren Sie das Modul und wählen Sie "Antriebsobjekt aktivieren".
5. Dadurch werden die Konfigurationsdaten des neuen Moduls eingelesen.
6. Aktualisieren Sie die Antriebsdaten im Inbetriebnahmearchiv mit den neuen Konfigurationsdaten.

6.1.10 Beispiel: Parallelschaltung mit TM120

Anwendung: 4 parallel geschaltete Motoren

Beispiel-Topologie:



M1 ... M4	Motor 1 ... Motor 4 Je Motor sind 1 x KTY und 1 x (3 PTC in Reihenschaltung) angeschlossen.
SMx	Sensor Module (Motorgeber)
ALM	Active Line Module
DMM	Double Motor Module
SMM	Single Motor Module
TM120	Terminal Module

Für die dargestellte Topologie werden 4 KTY-Sensoren und 4 PTC-Sensoren benötigt:

- Pro Primärteil gibt es 1 KTY-Sensor (Temp-F) und 3 in Reihe geschaltete PTC-Sensoren (Temp-S)

Es werden zwei TM120 benötigt:

- Ein TM120 wird in Reihe zwischen Motor Module und Gebermodul SMx automatisch verschaltet.
- Ein TM120 wird direkt an ein Line Module gesteckt: Hier ist eine manuelle Verschaltung durch den Inbetriebsetzer notwendig.

Maßnahmen am TM120:

1. **TM120 zwischen Motor Module und Gebermodul SMx**

   Mit diesem TM120 werden 4xKTY ausgewertet => Sensortypen müssen per Servo-p4610/TM-p4100 ausgewählt werden. Die zugehörige Temperatur wird per Servo-r4620/TM120-r4105 ausgegeben.

2. **TM120 direkt an Line Module**

   Mit diesem TM120 werden 4xPTC ausgewertet => Sensortypen müssen per TM-p4100 ausgewählt werden. Die zugehörige Temperatur wird per Servo-r4105 ausgegeben.

   Einstellung der Schwellwerte im Kontext PTC:

   – TM120-p4102[x]=251 => Auswertung aus

   – TM120-p4102[x]=120 => Auswertung ein

     Zuordnung der TM120-Reaktion mittels Störungspropagierung an den Antrieb => Einstellung der Propagierung per Servo-p0609=BICO:<Objektnummer>TM120:4105.0

Zuordnung der Alarme zum Motor

Die auftretenden Fehlermeldungen werden den Motoren wie folgt zugeordnet:

- Temperaturkanal auf TM120 in Reihenschaltung mit Motor Module und Geber:

  Alarm 207015 <Ortsangabe>Antrieb: Motortemperatursensor Warnung

  Alarm 207016 <Ortsangabe>Antrieb: Motortemperatursensor Störung

  Alarm 235920 <Ortsangabe>TM: Fehler Temperatursensor Kanal 0

  In diesem Fall wird eine Meldung mit Bezug zum Motor und zur Temperaturkomponente ausgegeben, aus der der betroffene Motor ersichtlich ist.

- Temperaturkanal auf TM120 direkt an Line Module:

  Alarm 235207 <Ortsangabe>TM: Temperatur Stör-/Warnschwelle Kanal 0 überschritten

  Besonderheit bei PTC:

  TM120-r4105 = -50 Temperatur ist unterhalb der Nennansprechtemperatur

  TM120-r4105 = 250 Temperatur ist oberhalb der Nennansprechtemperatur

  In diesem Fall wird eine Meldung nur mit Bezug zur Temperaturkomponente ausgegeben.

  Aus der Information zum TM120 kann der betroffene Motor ermittelt werden.

6.1.11 Austausch von Prozessdaten

Logische E/A-Adresse des Antriebs

Über die E/A-Adresse wird der NC der Datenbereich bekannt gegeben, über den der zyklische Prozessdatenaustausch mit dem Antrieb und der Control Unit erfolgt:

- E/A-Adresse der Antriebe
- E/A-Adresse der Control Units
- Telegramme

Die im PLC-Projekt parametrierten E/A-Adressen werden in folgendes Maschinendatum eingetragen:

Maschinendatum		ohne NX-Baugruppe
13120[0]	$MN_CONTROL_UNIT_LOGIC_ADRESS	= 6500
13120[1]	$MN_CONTROL_UNIT_LOGIC_ADRESS	= 0

Maschinendatum		mit NX-Baugruppe
13120[0]	$MN_CONTROL_UNIT_LOGIC_ADRESS	= 6500
13120[1]	$MN_CONTROL_UNIT_LOGIC_ADRESS	= 6516

PROFIBUS-Telegrammtyp

Die Vorbelegung des Antriebstelegramms auf den PROFIBUS-Telegrammtyp 136 (Siemens Telegrammtyp wie 116 zzgl. Momentenvorsteuerung) findet im vorprojektierten SDB statt. Bei der Inbetriebnahme wird dieser Telegrammtyp in den Komponenten NCK (MD13060) und SINAMICS S120 (p0922) eingestellt.

Der eingestellte Systemdatenbaustein wird über MD11240[0...3] ausgewählt:

MD11240	$MN_PROFIBUS_SDB_NUMBER Nummer des verwendeten Systemdatenbausteins (SDB) zur Konfiguration der Peripherie.
MD11240[0] = 0...n	PROFIBUS
MD11240[1] = -1	Diese Voreinstellung ist zwingend beizubehalten.
MD11240[2] = 0...n	in SINAMICS S120 integrierter PROFIBUS
MD11240[3] = 0...n	PROFINET

Mit der Einstellung von MD11240[0..3] = **n** projektieren Sie folgende Eigenschaften:

Eigenschaft		PPU24x.2	PPU26x.2	PPU28x.2
Bustakt (Systemtakt)		1,5 ms	1,5 ms	1,5 ms
SERVO Takt		3 ms	3 ms	TE: 1,5 ms ME: 3 ms
n=0,1	Achsen CU-I	6x Telegramm 136	6x Telegramm 136	6x Telegramm 136
	Achsen NX10	3x Telegramm 136	3x Telegramm 136	3x Telegramm 136
n=3	Achsen CU-I	6x Telegramm 116	6x Telegramm 116	6x Telegramm 116
	Achsen NX10	3x Telegramm 116	3x Telegramm 116	3x Telegramm 116

Literatur

Die Beschreibung der Telegramme finden Sie in folgender Literatur:

- SINAMICS S120 Funktionshandbuch /FH1/, Kapitel "Kommunikation"
- SINAMICS S120 Listenhandbuch /LH1/ unter "Funktionspläne"

6.1.12 Parameter für den Testlauf Achse / Spindel

Relevante Parameter und Klemmen

Antrieb:

Parameter/Klemme	Bedeutung
p0840	EIN/AUS1
p0844	1. AUS2
p0845	2. AUS2
p0848	1. AUS3
p0849	2. AUS3
p0852	Betrieb freigeben
X21.3 (+24 V) und X21.4 (Masse)	EP-Klemmen Freigabe (Impulsfreigabe)
p0864	Einspeisung Freigabe
p1140	Hochlaufgeber Freigabe
p1141	Hochlaufgeber Start
p1142	Sollwert Freigabe

Einspeisung:

Parameter/Klemme	Bedeutung
p0840	EIN/AUS1
p0844	1. AUS2
p0845	2. AUS2
p0852	Betrieb freigeben
X21.3 (+24 V) und X21.4 (Masse)	EP-Klemmen Freigabe (Impulsfreigabe)

Die Antriebparameter werden gesetzt unter: Bedienbereich "Inbetriebnahme" → "Softkey "Antriebsgeräte" → "Eingänge/Ausgänge".

Siehe auch

Weitere Literatur zum Antrieb:

- Inbetriebnahmehandbuch SINAMICS S120
- Gerätehandbuch Leistungsteile Booksize

6.2 Achsen zuordnen

6.2.1 Beispiel: So ordnen Sie die Achsen zu

Achsen zuordnen

Nach dem abschließenden Speichern der Konfigurationsdaten von Geber 2, wird folgende Übersicht angezeigt:

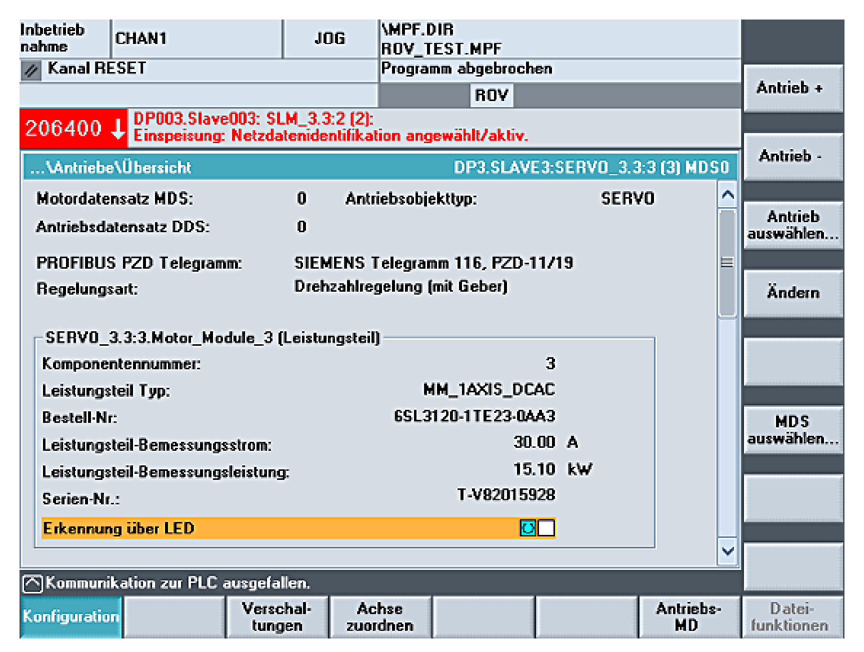


Bild 6-61 Antriebe\Übersicht

1. Um jedem logischen Antrieb und eine reale Achse zuzuordnen, wählen Sie den horizontalen Softkey "Achse zuordnen". Der Dialog "Achsenzuordnung" wird geöffnet:

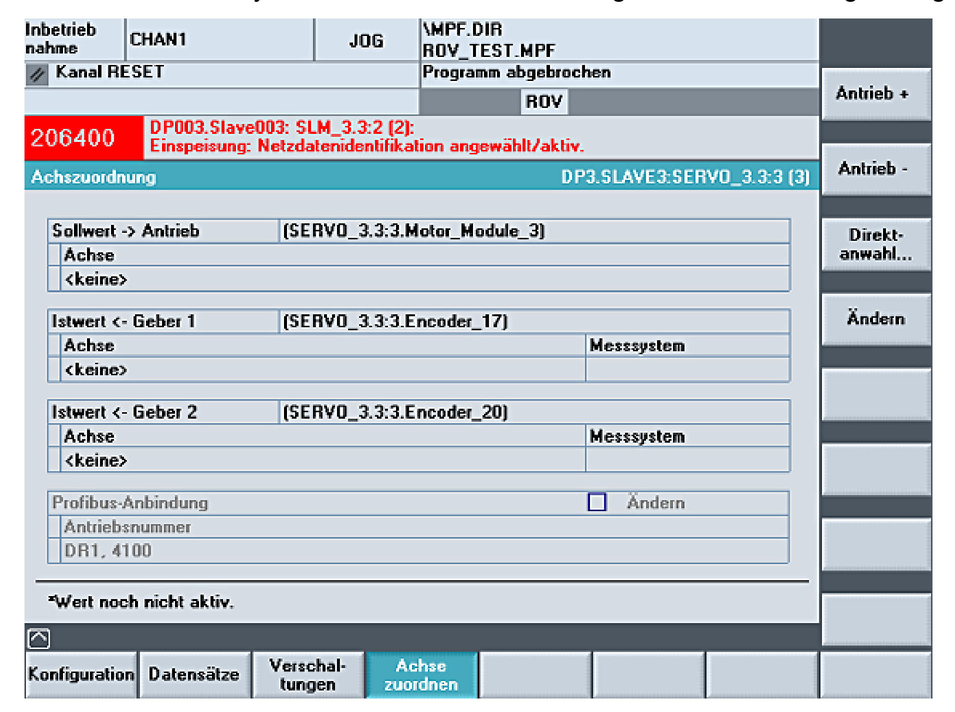


Bild 6-62 Achszuordnung

Hinweis

Die beiden folgenden Aktionen werden jeweils **zweimal** durchgeführt, d.h. die Softkeys "Ändern" und "Übernahme" müssen mit einer Wiederholung betätigt werden.

---

2. Drücken Sie den Softkey "Ändern", um DO Nummer 2 eine Achse zuzuordnen.
3. Wählen Sie aus der Auswahlliste "MSP1" und drücken Sie den Softkey "Übernahme".
4. Damit die Zuordnung im System wirksam wird, ist ein NCK Power On Reset notwendig.

5. Drücken Sie den Softkey "Abbruch", um zunächst die weiteren Achsen zuzuordnen.

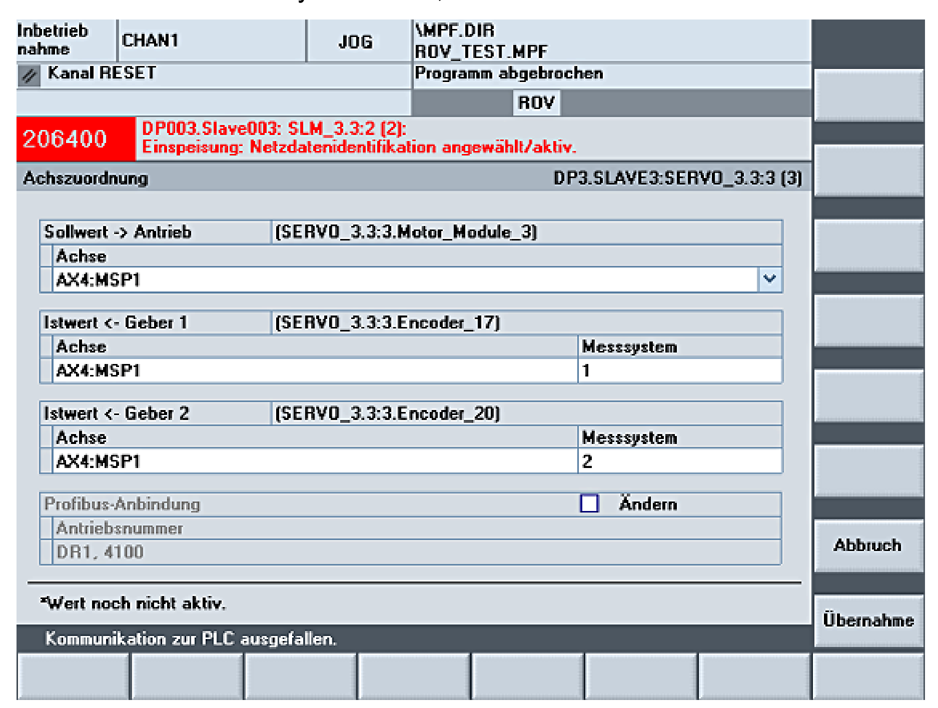


Bild 6-63 Achszuordnung: Spindel

6. Mit den vertikalen Softkeys "Antrieb +" und "Antrieb --" wählen Sie das nächste Modul an.
7. Ordnen Sie nacheinander alle Achsen zu:

Achse	Antrieb
MSP1	SERVO_3.3:3
MX1	SERVO_3.3:4
MY1	SERVO_3.3:5
MZ1	SERVO_3.3:6

Einstellungen übernehmen

Abschließend führen Sie einen NCK Power On Reset durch und überprüfen die folgenden Einstellungen:

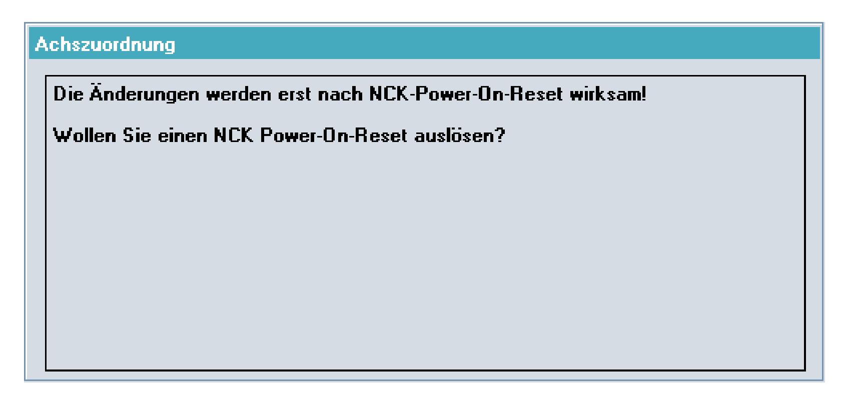


Bild 6-64 Achszuordnung: Power On Reset

Bestätigen Sie mit "OK" und führen Sie sowohl bei den Antrieben als auch bei der Steuerung einen Neustart aus. Auf dem PG/PC wird folgende Zuordnung angezeigt:

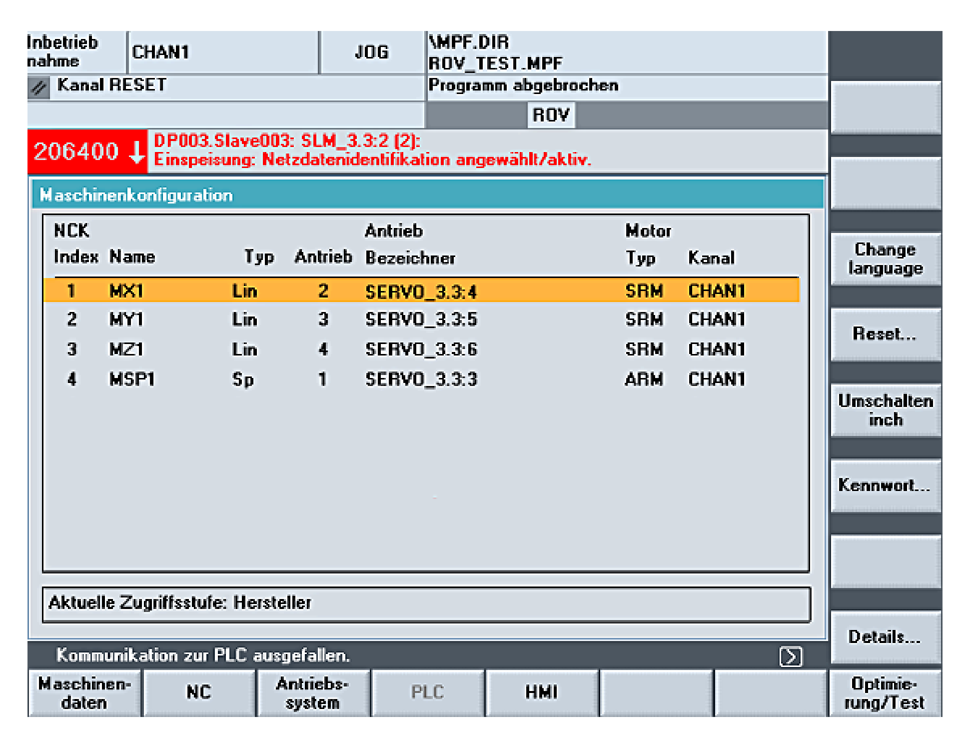


Bild 6-65 Maschinenkonfiguration

Nach dem Neustart wird an der Steuerung folgende Maschinenkonfiguration angezeigt:

JOG 206400 DP003.Slave003: SLM_3.3:2 (2)":
Einspeisung: Netzdatenidentifikation angewählt/ aktiv. 23.03.11 11:32

Maschinenkonfiguration

Maschinenachse			Antrieb		Motor	
Index	Name	Typ	Nr.	Bezeichner	Typ	Kanal
1	MX1	Linear	2	SERVO_3.3:4	SRM	CHAN1
2	MY1	Linear	3	SERVO_3.3:5	SRM	CHAN1
3	MZ1	Linear	4	SERVO_3.3:6	SRM	CHAN1
4	MSP1	Spindel S1	1	SERVO_3.3:3	ARM	CHAN1

Aktuelle Zugriffsstufe: Hersteller

Change language | Reset (po) | Kennwort | Details | Daten sichern

Masch. daten | NC | Antriebs system | PLC | HMI | System daten | Optimie. Test

Bild 6-66 Maschinenkonfiguration: Alle Achsen in Betrieb genommen

Ergebnis

Schritt 5 zur Inbetriebnahme des Antriebssystems ist damit abgeschlossen.

Datensicherung

Mit dem vertikalen Softkey "Daten sichern" werden die Konfigurationsdaten nach der Inbetriebnahme im nicht-flüchtigen Speicher gesichert:

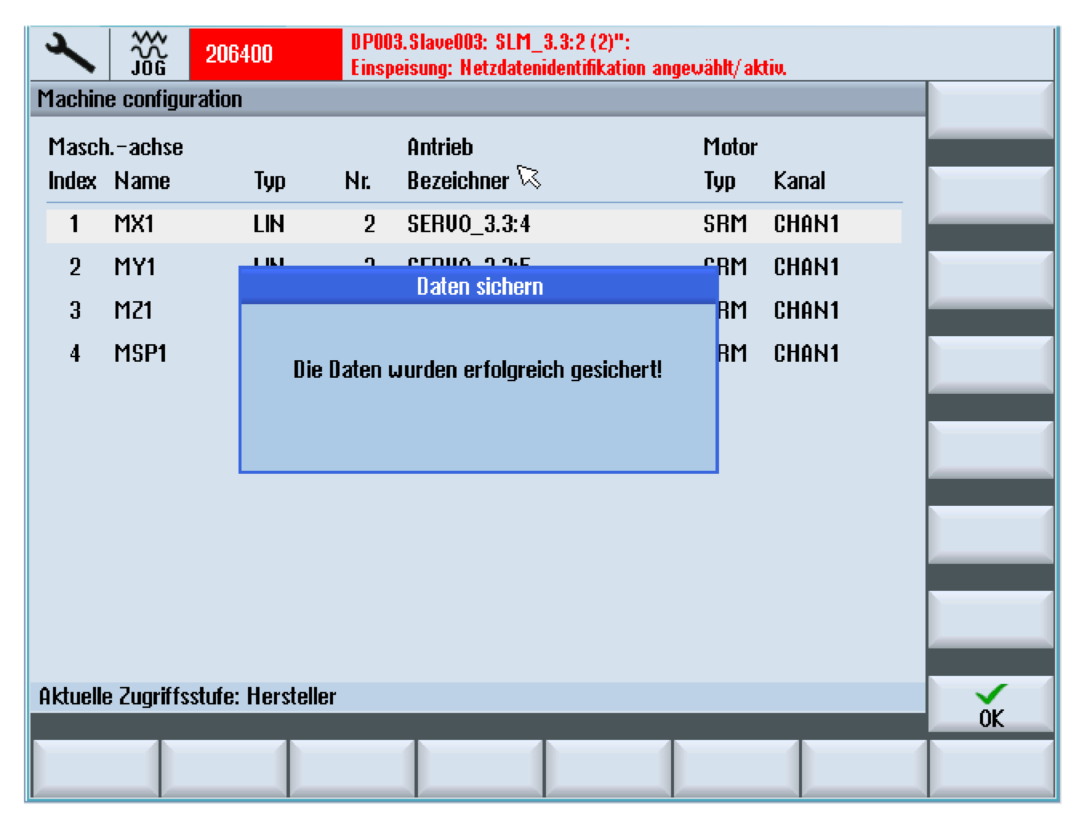


Bild 6-67 Meldung zur Datensicherung

6.2.2 Beispiel: Maschinendaten für Achse/Spindel einstellen

Achs-Maschinendaten

Nach der Inbetriebnahme im vorangegangenen Beispiel sind bei den Achs-Maschinendaten die folgenden Werte eingestellt:

Achs-Maschinendaten		X	Y	Z	SP	A
30200	$MA_NUM_ENCS	1	1	1	1	1
30230	$MA_ENC_INPUT_NR	1	1	1	2	1
30240	$MA_ENC_TYPE	1	1	1	1	4
31020	$MA_ENC_RESOL	2048	2048	2048	1024	512
34200	$MA_ENC_REFP_MODE	1	1	1	1	0

Spindel-Maschinendaten

Um die Spindel im vorangegangenen Beispiel nach der Inbetriebnahme des Antriebs im JOG Betrieb zu verfahren, geben Sie bei den folgenden Maschinendaten die neuen Werte ein:

Spindel-Maschinendaten		Voreinstellung	Neuer Wert
32000	$MA_MAX_AX_VELO	10000	3000
32010	$MA_JOG_VELO_RAPID	10000	100
32020	$MA_JOG_VELO	2000	50
35100	$MA_SPIND_VELO_LIMIT	10000	3000
35110[0]	$MA_GEAR_STEP_MAX_VELO[0]	500	3000
35110[1]	$MA_GEAR_STEP_MAX_VELO[1]	500	3000
35130[0]	$MA_GEAR_STEP_MAX_VELO_LIMIT[0]	500	3150
35130[1]	$MA_GEAR_STEP_MAX_VELO_LIMIT[1]	500	3150
36200[0]	$MA_AX_VELO_LIMIT[0]	11500	3300
36200[1]	$MA_AX_VELO_LIMIT[1]	11500	3300

Antriebsparameter Maximaldrehzahl

Nach DDS-Umschaltung von z. B. DDS0 → DDS1 kann die Spindel lediglich mit der Drehzahl aus p1082[0] gefahren werden. Wird in p1082[0] ein kleinerer Wert als in p1082[1...n] eingegeben, wird die Spindeldrehzahl auf den Wert in p1082[0] begrenzt.

Hinweis

Im Parameter p1082[0] ist die maximale Drehzahl einzutragen:

- Wenn der Parameter mit einem größeren Wert als in den übrigen Parametern besetzt wird, ist das Verfahren der Spindel mit maximaler Drehzahl möglich.
- Wenn der Parameter p1082[0] nachträglich auf einen kleineren Wert geändert wird, und anschließend ein NCK-Reset oder Power OFF/ON ausgeführt wird, dann verfährt die Spindel wieder ausschließlich auf den kleineren Wert aus p1082[0] und alle anderen Werte werden ignoriert.

Diagnose-Anzeigen

Weitere Informationen zum Status von Achsen und Spindel erhalten Sie im Bedienbereich "Diagnose" mit der Menüfortschalt-Taste:

- Der Softkey "Achs-Diagnose" öffnet die "Service Übersicht".
- Der Softkey "Service Achse" öffnet den Dialog "Service-Achse/Spindel".

Hinweis

DSC-Betrieb

Die Parametrierung der Kombination STIFFNESS_CONTROL_ENABLE=1 und ENC_FEEDBACK_POL= -(Geber-Invertierung) ist nicht erlaubt.

Die Geber-Invertierung muss im Antriebsparameter p0410 Bit 1 des SINAMICS erfolgen.

- DSC-Betrieb ist für Motor-Messsysteme voreingestellt.
- DSC-Betrieb für externe Messsysteme muss explizit aktiviert werden.

  Voraussetzung: Telegramm ≥ 116

  Antriebsparameter SINAMICS:

  p1192[0] Geberauswahl

  p1193[0] Geberanpassungsfaktor

6.3 Datensätze konfigurieren

6.3.1 Datensätze - Übersicht

Voraussetzung

Hinweis

Bei den jeweiligen Antrieben muss eine Inbetriebnahme erfolgt sein.

Datensätze

Die Konfiguration der Datensätze erfolgt im Bedienbereich "Inbetriebnahme" → Antriebssystem" → "Antriebe" → "Datensätze". Durch folgende Abläufe werden Sie Schritt für Schritt geführt:

- "Datensatz hinzufügen"
- "Datensatz entfernen", wenn bereits ein weiterer Datensatz angelegt wurde.
- "Datensatz modifizieren"

Anzahl der konfigurierbaren Datensätze:

- Motordatensatz → MDS0...3 (max. 4)
- Antriebsdatensatz → DDS0...31 (max. 8 pro MDS)
- Geberdatensätze → EDS0...2 (max. 3)

Hinweis

Antriebsparameter mit Auswirkungen auf Datensätze

Voreinstellung p2038 = 0 für Interface Mode: SINAMICS

Bei Auswahl eines Telegramms über p0922 wird über den Parameter p2038 die gerätespezifische Belegung der Bits in den Steuer- und Zustandsworten beeinflusst.

Bei p0922 = 100 ... 199 wird automatisch p2038 = 1 gesetzt und das Ändern von p2038 gesperrt. Damit ist bei diesen Telegrammen unveränderlich der Interface Mode "SIMODRIVE 611U" mit 8 DDS pro MDS eingestellt.

Anwendung

Der Data Set Assistant (DSA) bietet eine einfache und effiziente Lösung für Maschinenhersteller (OEM), Daten zu duplizieren und zu modifizieren, um das Verhalten von Antrieben, Motoren und Gebern zu konfigurieren. Die Wiederverwendung von Daten im DSA hilft dabei, den benötigten Konfigurationsaufwand auf ein Minimum zu reduzieren, wenn es darum geht, Maschinendaten für die Abstimmung der verschiedenen Teile zu generieren.

Die Bauteilmerkmale des Antriebssystems werden über Motor Data Sets (MDS), Drive Data Sets (DDS) sowie Encoder Data Sets (EDS) konfiguriert. Die Wiederverwendung von Daten reduziert die Konfigurationsstufen, die für die Abstimmung der Maschinendynamik der einzelnen Teile erforderlich sind.

Literatur

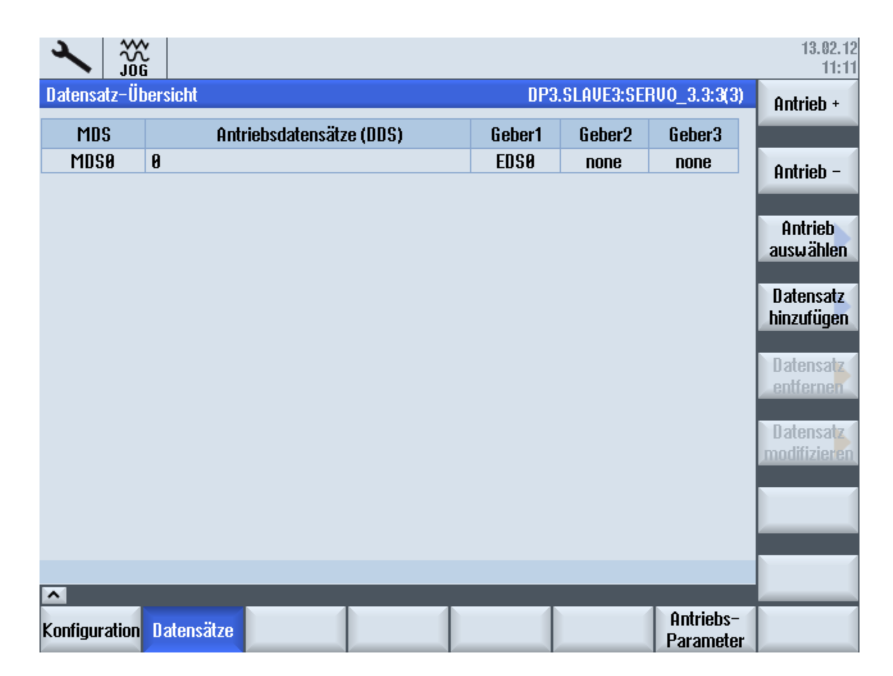
Weiterführende Informationen finden Sie in:

SINUMERIK 840D sl/828D Funktionshandbuch Grundfunktionen, Diverse NC/PLC-Nahtstellensignale und Funktionen (A2)

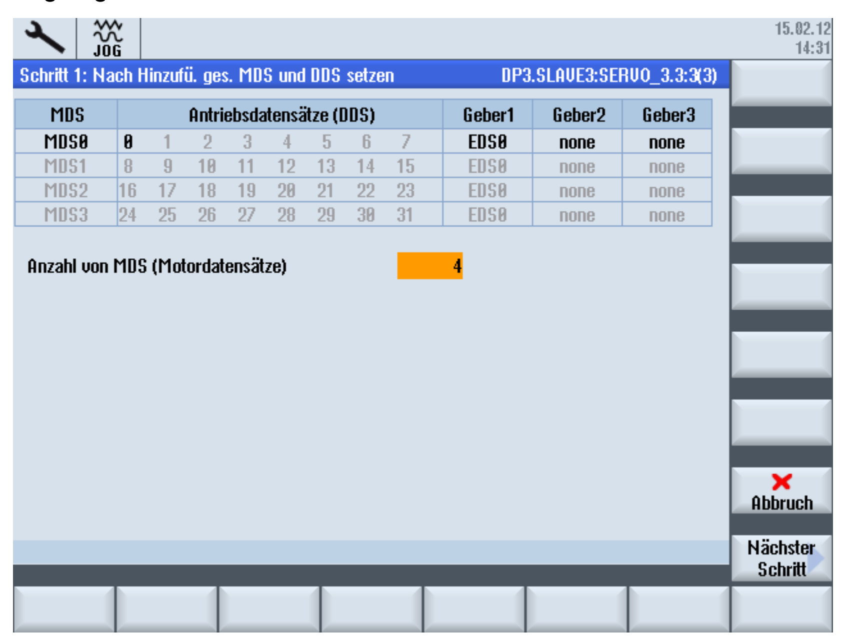
6.3.2 Datensatz hinzufügen

Bedienfolge Datensatz hinzufügen

Bei Werkseinstellung ist die Voreinstellung ein Motordatensatz MDS0 mit einem Antriebsdatensatz DDS0 und einem Geberdatensatz EDS0:



1. Drücken Sie "Datensatz hinzufügen". Im Beispiel wird die maximale Anzahl von 4 MDS angelegt.



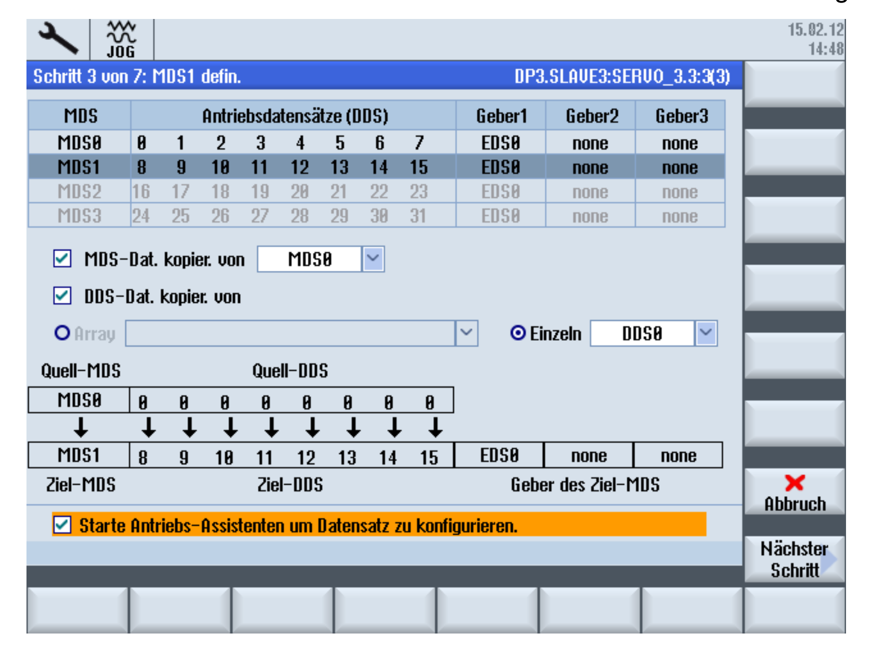
2. Drücken Sie "Nächster Schritt >".

   Im ersten Schritt werden im MDS0 die Antriebsdatensätze DDS1 bis DDS7 angelegt.

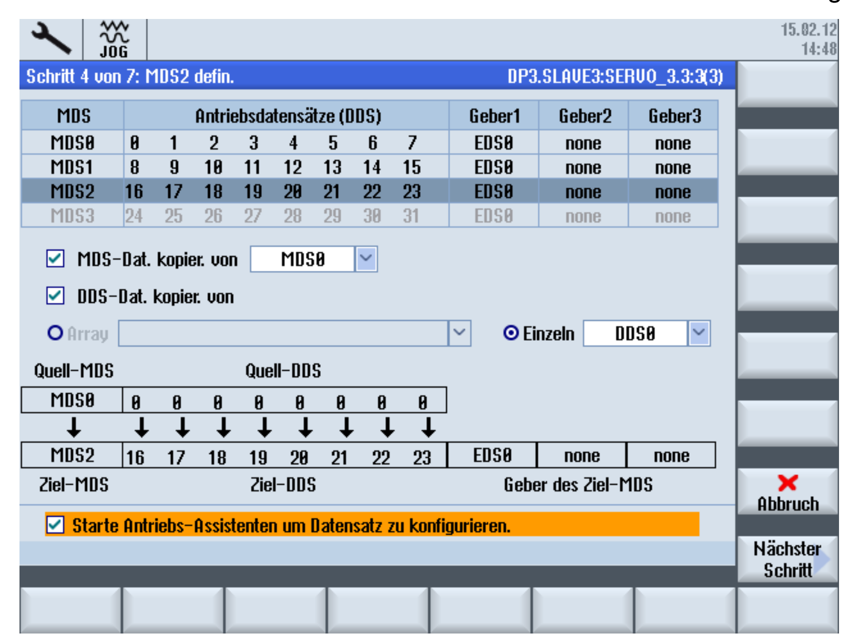




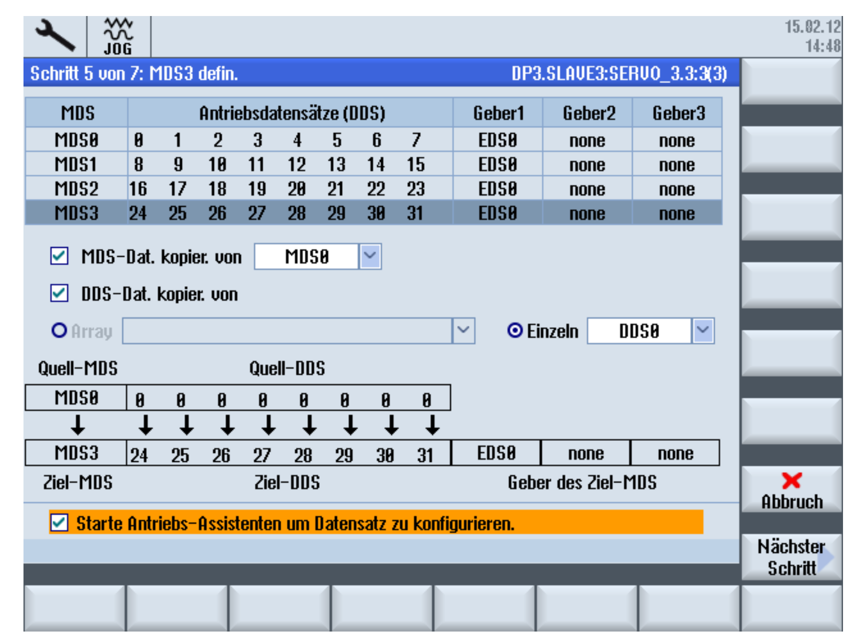
3. Drücken Sie "Nächster Schritt >". Der Motordatensatz MDS1 wird angelegt.



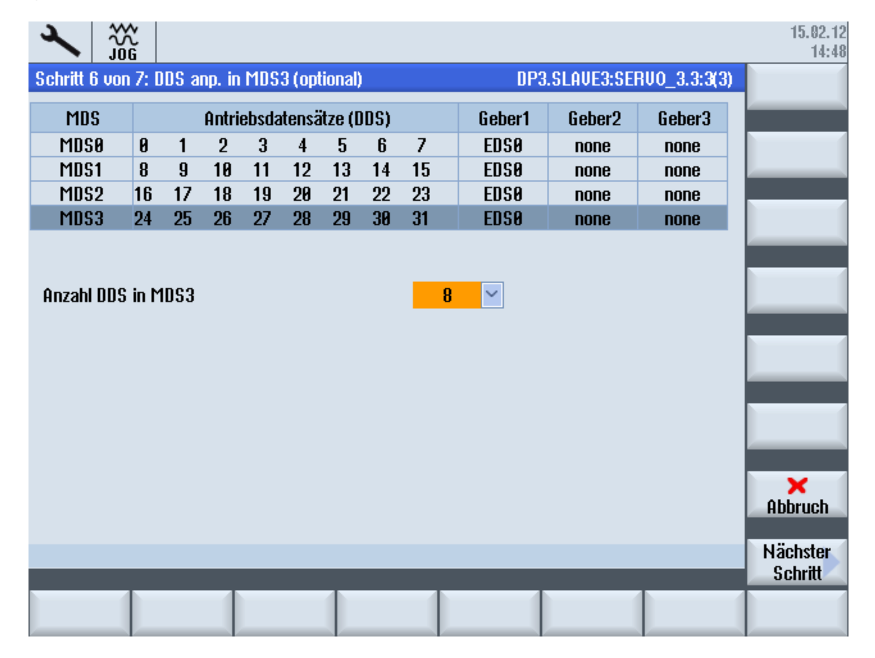
4. Drücken Sie "Nächster Schritt >". Der Motordatensatz MDS2 wird angelegt.



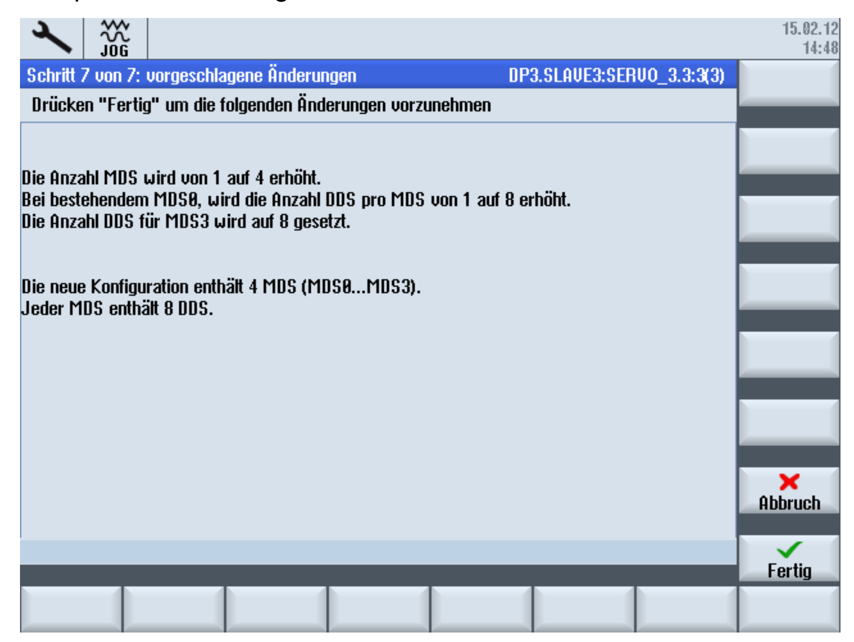
5. Drücken Sie "Nächster Schritt >". Der Motordatensatz MDS3 wird angelegt.



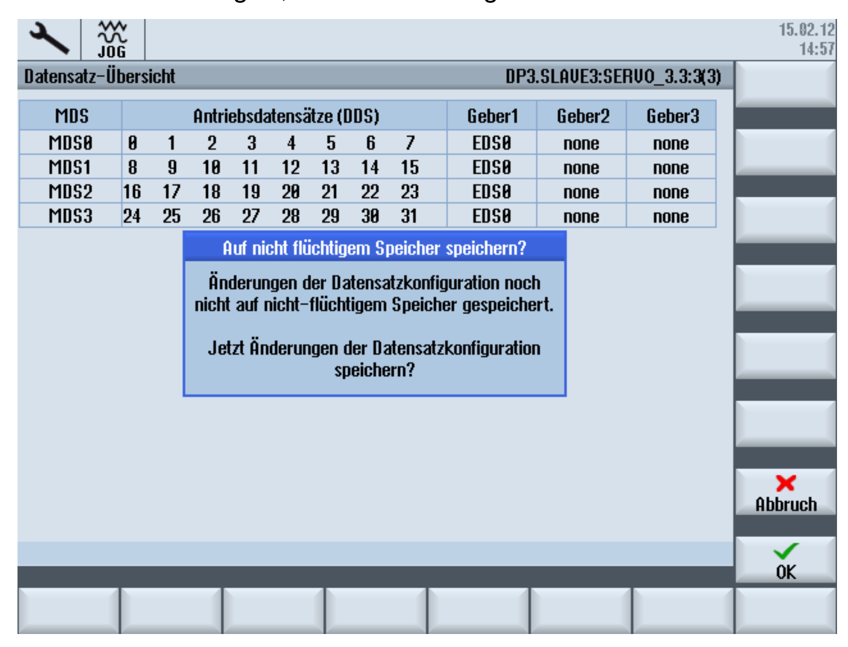
6. Drücken Sie "Nächster Schritt >". Erstellen Sie jeweils einen vollständigen MDS wie in der folgenden Abbildung oder geben Sie hier optional die Anzahl der DDS für den MDSx ein:



7. Drücken Sie "Nächster Schritt >", um in der Zusammenfassung die Änderungen zu überprüfen, die durchgeführt werden sollen.



8. Drücken Sie "Fertig >", um die Änderungen zu übernehmen.

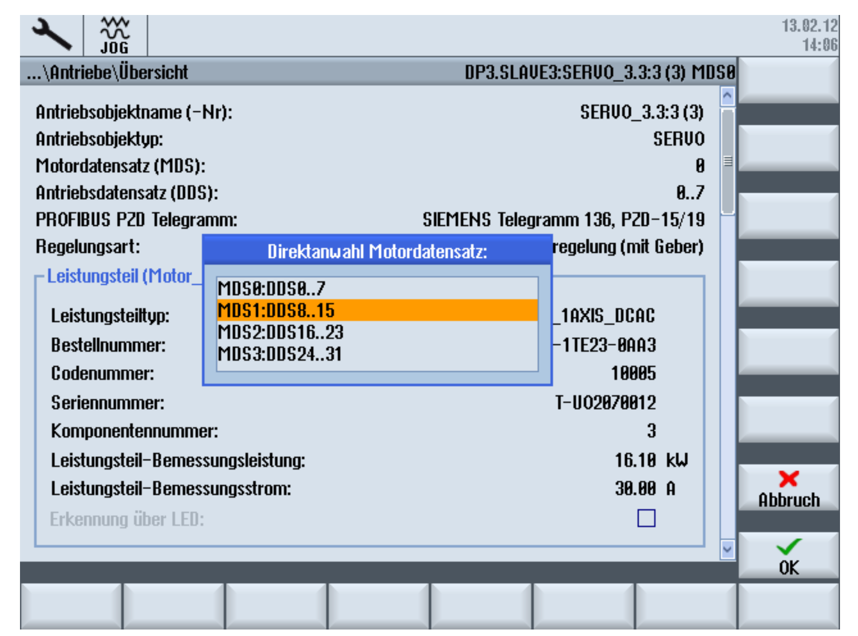


9. Bestätigen Sie mit "OK", um die Daten nicht-flüchtig zu speichern.

   Das Speichern/Schreiben der Parameter kann einige Minuten dauern.

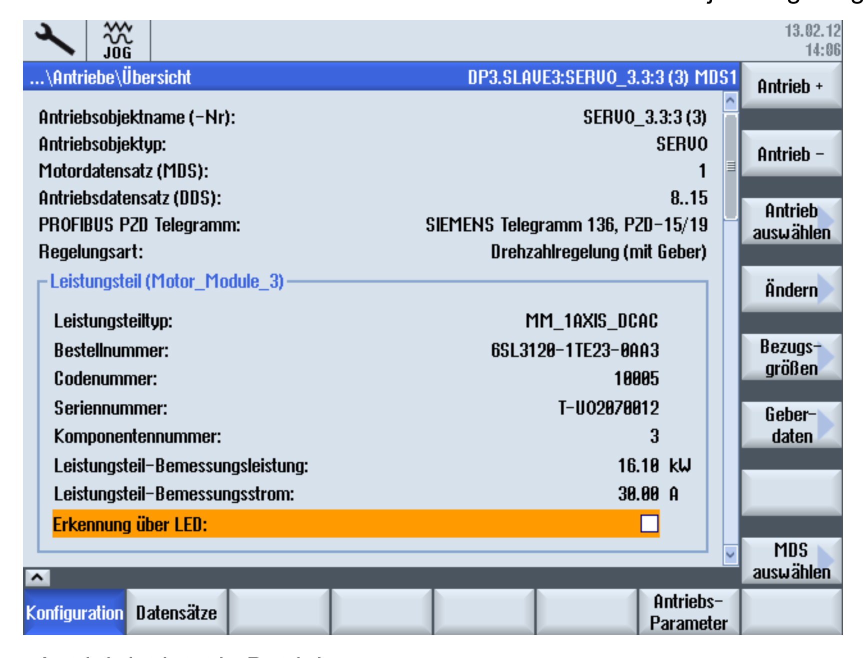


10. Der Dialog "Antriebe" → "Übersicht" des Antriebsobjekts wird angezeigt. Der Softkey "MDS auswählen >" ist nun bedienbar. Drücken Sie "MDS auswählen...".



In der Direktanwahl wählen Sie den Motordatensatz, hier z. B. MDS1.

11. Mit "OK" wird der neue Motordatensatz für das Antriebsobjekt angezeigt.



Der Antrieb ist jetzt in Betrieb genommen.

6.3.3 Datensatz entfernen

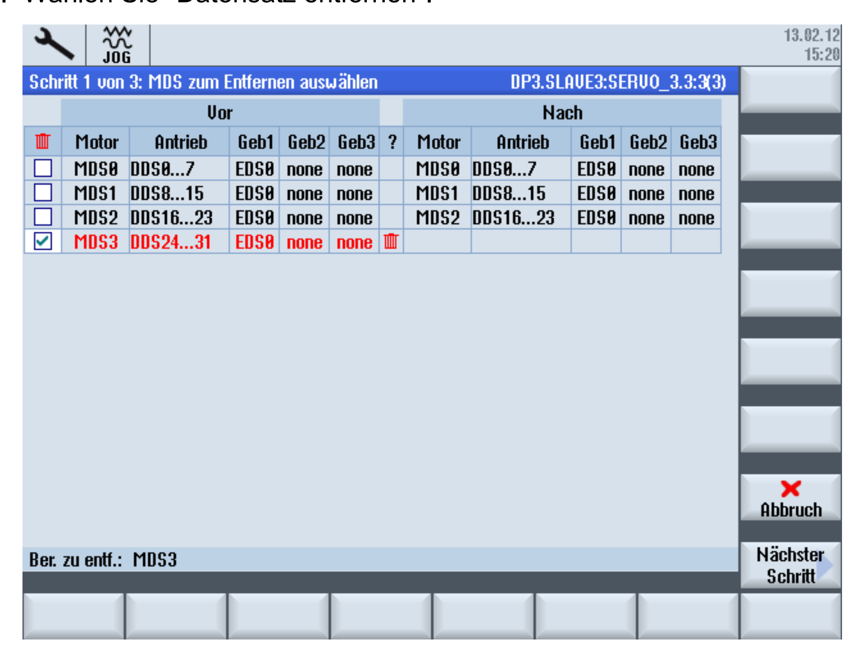
Voraussetzung

Um Datensätze zu entfernen, sind folgende Voraussetzungen notwendig:

- Anzahl der DDS > 1 im MDS ⇒ es können DDS entfernt werden.
- Anzahl der MDS > 1 ⇒ es können MDS entfernt werden.

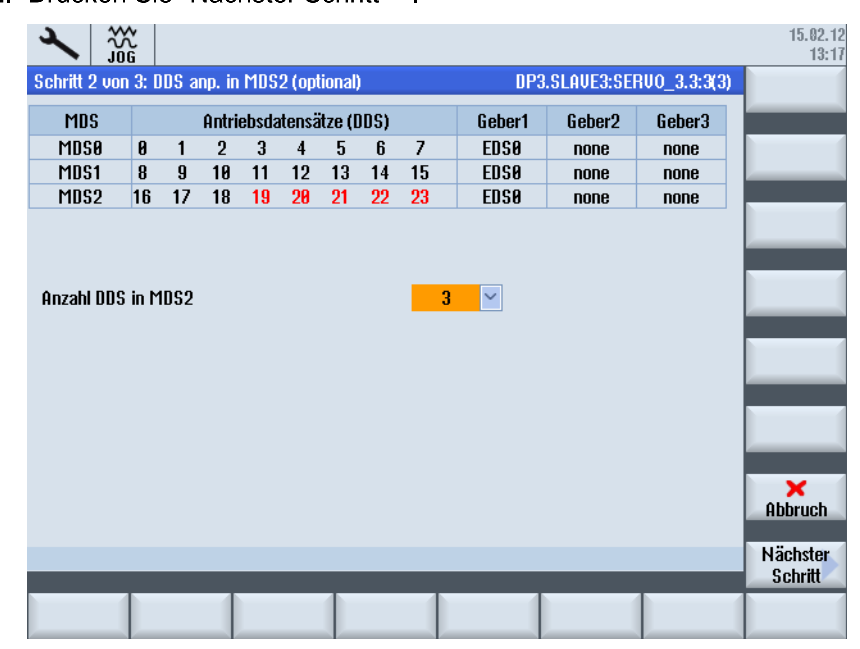
Bedienfolge Datensatz entfernen

1. Wählen Sie "Datensatz entfernen".



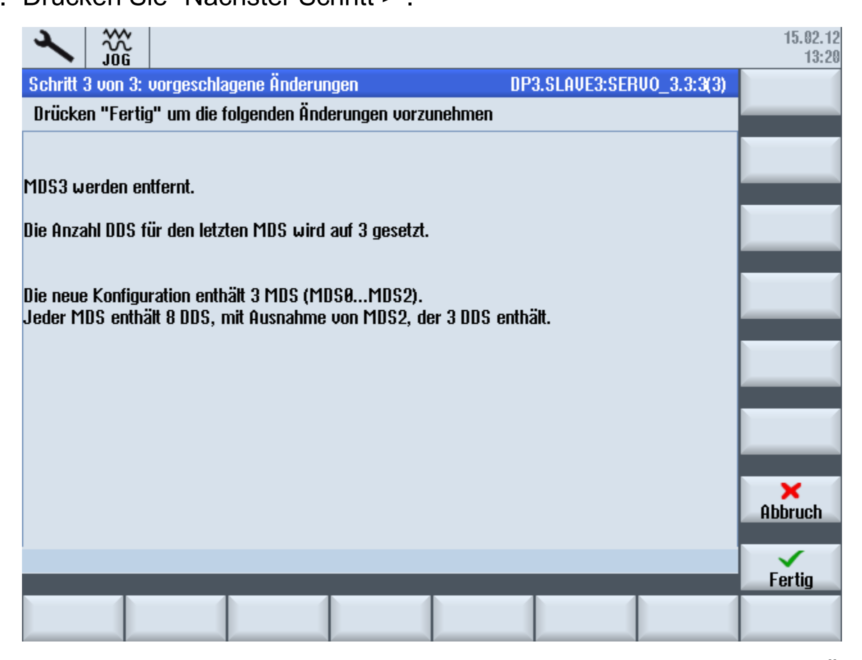
In der ersten Spalte wählen Sie den MDS, der entfernt werden soll, indem Sie die Zeile mit einem Häkchen markieren. Es können auch mehrere Datensätze ausgewählt werden.

2. Drücken Sie "Nächster Schritt >".



Optional entfernen Sie durch die Eingabe der Anzahl der Antriebsdatensätze DDS, die im MDS bleiben sollen, die restlichen DDS aus MDS2: Durch die Eingabe von 3 DDS in MDS werden 5 DDS entfernt.

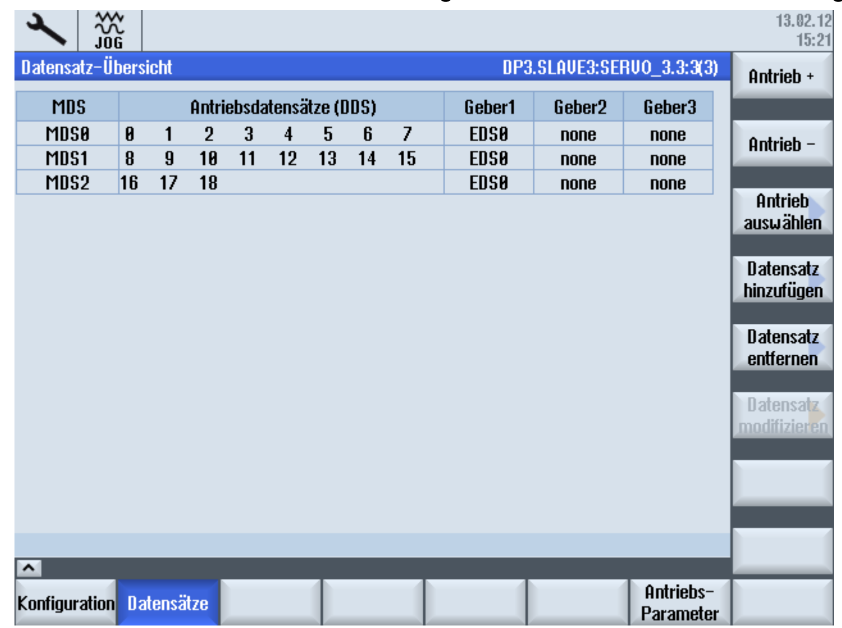
3. Drücken Sie "Nächster Schritt >".



Mit "Abbruch" wird dieser Vorgang abgebrochen und die Datensatz-Übersicht mit MDS0 ... MDS3 angezeigt.

4. Drücken Sie "Fertig >", um die Änderungen zu übernehmen.

   Die Daten werden in die Parameter geschrieben. Danach wird das Ergebnis angezeigt:



Beim Verlassen dieses Dialogs bestätigen Sie die Anfrage mit "Ja", um die Daten nichtflüchtig zu speichern.

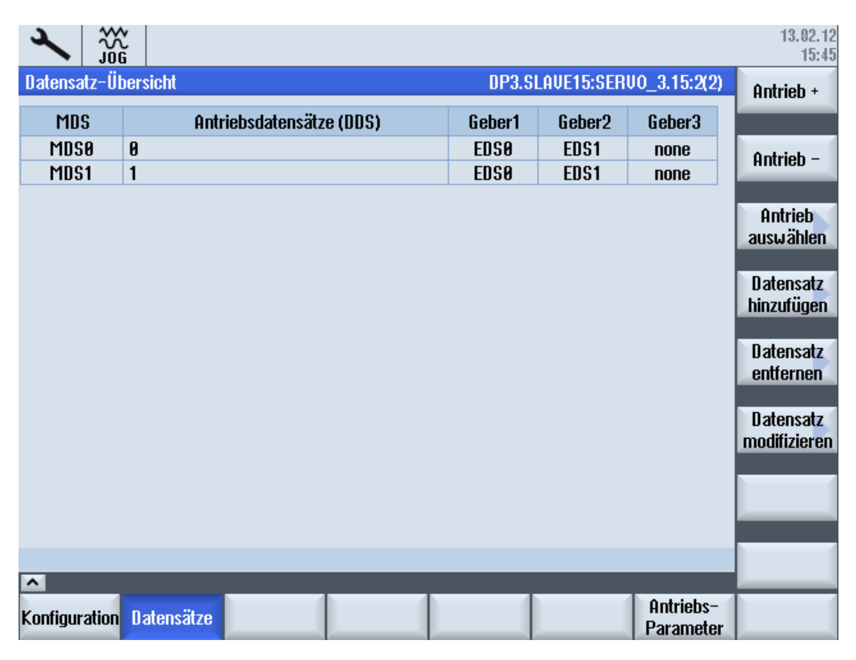
6.3.4 Datensatz modifizieren

Voraussetzung

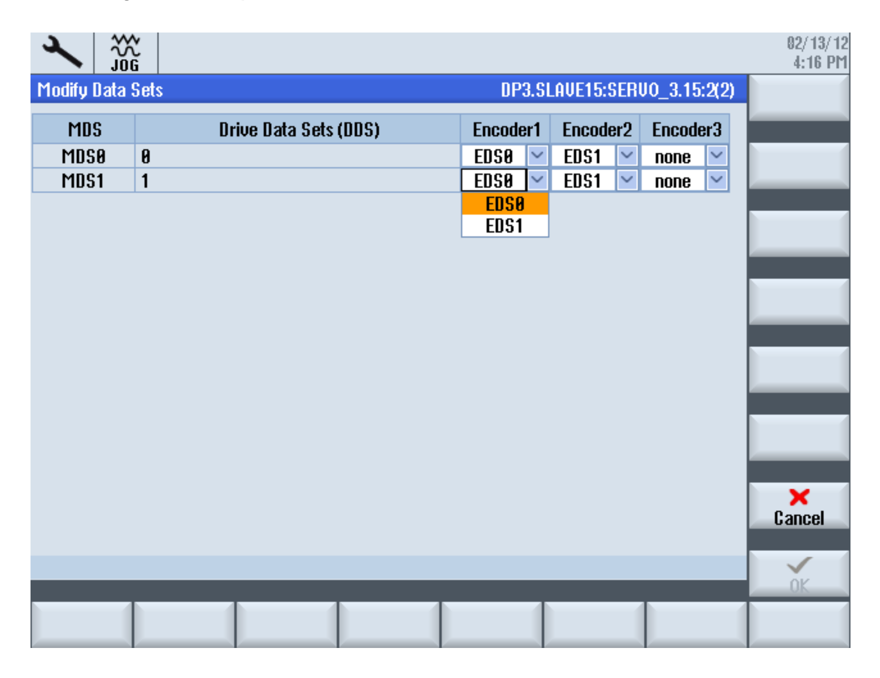
Sind dem Antriebsobjekt (SERVO) mehr als ein Geber zugeordnet, werden zusätzliche Geberdatensätze (EDS0...2) angelegt.

Datensatz modifizieren

Um die Zuordnung dieser Geberdatensätze zum jeweiligen Geber zu ändern, wählen Sie den Softkey "Datensätze modifizieren":



Damit aktivieren Sie die Editierfunktion, um die Zuordnung der EDS zu den Gebern nachträglich anzupassen:



6.4 Topologieregeln für DRIVE-CLiQ

6.4.1 Topologieregeln für S120 Combi

Topologieregeln für DRIVE-CLiQ

Für den S120 Combi bestehen feste DRIVE-CLiQ-Topologieregeln. Diese Regeln müssen eingehalten werden. Eine Verletzung dieser Regeln wird mit einer entsprechenden Warnung angezeigt.

Belegung der DRIVE-CLiQ- Schnittstellen

Tabelle 6- 1 Belegung der DRIVE-CLiQ-Schnittstellen am S120 Combi

DRIVE-CLiQ-Schnittstelle	Verbindung mit
X200	X100 der PPU
X201	Motorgeber Spindel
X202	Motorgeber Vorschub 1
X203	Motorgeber Vorschub 2
X204	Motorgeber Vorschub 3 → nur bei 4 Achsen S120 Combi bleibt leer bei 3 Achsen S120 Combi
X205	Optional: 2. direkter sin/cos-Geber für Spindel (über SMx20) 1) bleibt leer bei Anschluss eines direkten TTL-Spindelgebers über X220

1) in diesem Fall bleibt die TTL-Geberschnittstelle X220 frei

Tabelle 6- 2 Belegung der DRIVE-CLiQ-Schnittstellen an der SINUMERIK 828D (PPU)

DRIVE-CLiQ-Schnittstelle	Verbindung mit
X100	X200 des S120 Combi
X101	X200 eines Single Motor Modules oder Double Motor Modules
X102	X500 des Terminal Module TM54F X500 des Hub Modules (DMx20) 1)

1) bei Verwendung eines TM54F wird das DMx20 über die DRIVE-CLiQ-Schnittstelle X501 am TM54F in Reihe geschaltet

Tabelle 6- 3 Belegung der DRIVE-CLiQ-Schnittstellen der Erweiterungsachsen

DRIVE-CLiQ-Schnittstelle	Verbindung mit
Erstes Single Motor Module	
X200	X101 der PPU
X201 [1]	X200 des zweiten Single Motor Modules
X202	Motorgeber für Vorschub 1. Erweiterungsachse (über Sensor Module)
Zweites Single Motor Module	
X200	X201 des ersten Single Motor Modules
X201	bleibt leer
X202	Motorgeber für Vorschub 2. Erweiterungsachse (über Sensor Module)
Double Motor Module	
X200	X101 der PPU
X201	bleibt leer
X202	Motorgeber für Vorschub 1. Erweiterungsachse
X203	Motorgeber für Vorschub 2. Erweiterungsachse

[1] bleibt leer, wenn nur ein Single Motor Module verwendet wird

Tabelle 6- 4 Belegung der DRIVE-CLiQ-Schnittstellen am DMx20 für die Zuordnung eines direkten Messsystems zu den Vorschubachsen

DRIVE-CLiQ-Schnittstelle	Vorschubachse
X500	X501 des TM54F X102 der PPU (wenn kein TM54F verwendet wird)
X501	Direkter Geber Vorschub 1 am S120 Combi
X502	Direkter Geber Vorschub 2 am S120 Combi
X503	Direkter Geber Vorschub 3 am S120 Combi → nur bei 4 Achsen S120 Combi bleibt leer bei 3 Achsen S120 Combi)
X504	Vorschub 1. Erweiterungsachse am Motor Module
X505	Vorschub 2. Erweiterungsachse am Motor Module

Tabelle 6- 5 Belegung der DRIVE-CLiQ-Schnittstellen am TM54F

DRIVE-CLiQ-Schnittstelle	
X500	X102 der Steuerung (PPU)
X501	X500 des DMx20 Wird kein DMx20 verwendet, bleibt diese Schnittstelle leer.

6.4.2 Topologieregeln für S120 Booksize

Einleitung

Für die Verdrahtung von Komponenten mit DRIVE-CLiQ gibt es folgende Regeln. Man unterscheidet zwischen solchen **DRIVE-CLiQ-Regeln**, die unbedingt eingehalten werden müssen, und **empfohlenen Regeln**, die, wenn sie eingehalten werden, bei Erweiterungen keine nachträglichen Änderungen der Topologie erfordern.

Die maximale Anzahl der DRIVE-CLiQ-Komponenten und die mögliche Art ihrer Verdrahtung ist abhängig von folgenden Punkten:

- den verbindlichen DRIVE-CLiQ-Verdrahtungsregeln
- der Anzahl und Art der aktivierten Antrieben und Funktionen auf der jeweiligen Control Unit
- der Rechenleistung der jeweiligen Control Unit
- den eingestellten Verarbeitungs- und Kommunikationstakten

Neben den verbindlichen Verdrahtungsregeln und einigen zusätzlichen Empfehlungen sind Beispiel-Topologien für DRIVE-CLiQ-Verdrahtungen im Gerätehandbuch PPU angegeben.

Gegenüber diesen Beispielen können Komponenten entfernt, gegen andere ausgetauscht oder ergänzt werden. Sofern Komponenten gegen einen anderen Typ ausgetauscht werden oder zusätzliche Komponenten hinzugefügt werden, sollte diese Topologie mit dem Tool SIZER überprüft werden.

Hinweis

Jede Topologie die der SIZER erlaubt, ist auch lauffähig und somit korrekt (SINAMICS S120 Funktionshandbuch /FH1/).

DRIVE-CLiQ-Regeln

Die nachfolgenden Verdrahtungsregeln gelten für Standardtaktzeiten (Servo 125 μs). Bei kürzeren Taktzeiten als den jeweiligen Standardtaktzeiten ergeben sich weitere Einschränkungen aus der Rechenleistung der Control Unit.

Die nachfolgenden Regeln gelten generell, sofern sie nicht eingeschränkt werden, abhängig von der Firmware-Version.

- Es sind maximal 8 DRIVE-CLiQ-Teilnehmer in einer Reihe zulässig. Eine Reihe wird immer von der Control Unit aus betrachtet.

  Diese Anzahl verringert sich auf maximal 4 DRIVE-CLiQ-Teilnehmer in einer Reihe, wenn die Extended Functions des antriebsbasierten Safety Integrated projektiert sind.
- Es sind maximal 14 DRIVE-CLiQ-Teilnehmer an einem DRIVE-CLiQ-Strang an einer Control Unit zulässig.

- Es ist keine Ringverdrahtung zugelassen.
- Die Komponenten dürfen nicht doppelt verdrahtet sein.

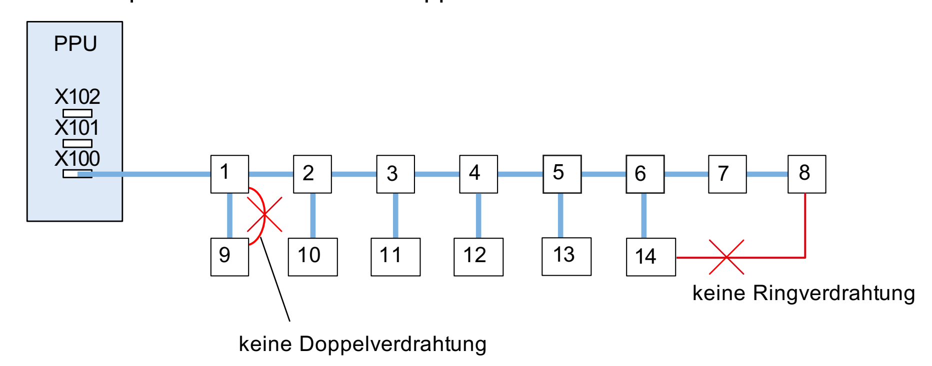


Bild 6-68 Beispiel: DRIVE-CLiQ-Strang an der Klemme X100 (ohne Safety Integrated)

- Für die Bauform Booksize gilt:
  - In den Betriebsarten Servoregelung und Vektor U/f-Steuerung darf nur genau ein Line Module an die Control Unit angeschlossen werden. In der Betriebsart Vektorregelung dürfen maximal drei weitere Line Modules dazu parallel geschaltet werden (also insgesamt 4 Line Modules).
  - Ein Line Module und Motor Modules dürfen in der Betriebsart Servoregelung an einen gemeinsamen DRIVE-CLiQ-Strang angeschlossen werden.
  - Ein Line Module und Motor Modules müssen in der Betriebsart Vektorregelung an getrennte DRIVE-CLiQ-Stränge angeschlossen werden.
  - Bei der Bauform Booksize ist eine Parallelschaltung von Infeed Modules oder Motor Modules nicht erlaubt.
- Für die Bauform Chassis gilt: Line Modules (Active Line, Basic Line, Smart Line) und Motor Modules müssen an getrennte DRIVE-CLiQ-Stränge angeschlossen werden.
- Das Ändern der voreingestellten Abtastzeiten ist zulässig.
- An einer PPU mit NX10.3 können insgesamt maximal 16 Messsysteme angeschlossen werden:

  Beispiel 1: PPUmit 6 Achsen mit 6 Motormesssystemen und 6 direkten Messsystemen sowie NX10.3 mit 2 Motormesssystemen und 2 direkten Messsystemen.

  Beispiel 2: PPU mit 5 Achsen mit 5 Motormesssystemen und 5 direkten Messsystemen sowie NX10.3 mit 3 Motormesssystemen und 3 direkten Messsystemen.
- Es ist pro PPU und pro NX10.3 nur ein TM54F zugelassen.

- Das Active Line Module Booksize und die Motor Modules Booksize können an einem DRIVE-CLiQ-Strang angeschlossen werden.
- Chassis Line Module und Motor Module werden in Reihe geschaltet.
- Damit folgende Module bei der Inbetriebnahme automatisch zugeordnet (Geräteerkennung) werden, sollten sie an einen freien DRIVE-CLiQ-Port des zugehörigen Active Line Module/Motor Module angeschlossen werden:
  - Voltage Sensing Module (VSM)
  - Terminal Module TM120

  Hinweis

  Wird das TM120 nicht gemäß dieser Regel angeschlossen, muss die Zuordnung der Temperaturkanäle zum Antrieb vom Inbetriebsetzer über BICO-Technik erstellt werden.

- Die Abtastzeiten (p0115[0] und p4099) aller Komponenten, die an einem DRIVE-CLiQ-Strang angeschlossen sind, müssen unter sich ganzzahlig teilbar sein. Wenn an einem DO die Stromregler-Abtastzeit in ein anderes Raster geändert werden muss, das nicht zu den anderen DO am DRIVE-CLiQ-Strang passt, sind folgende Möglichkeiten gegeben:
  - DO an einem separaten DRIVE-CLiQ-Strang umstecken.
  - Die Stromreglerabtastzeit oder die Abtastzeit der Ein-/Ausgänge der nicht betroffenen DO ebenfalls so ändern, dass sie wieder ins Raster passt.

  Hinweis

  Ein Double Motor Module, ein DMC20 und ein TM54F entsprechen jeweils zwei DRIVE-CLiQ-Teilnehmern. Dies gilt auch für Double Motor Modules, von denen nur ein Antrieb konfiguriert ist.

Damit die Funktion "Automatische Konfiguration" die Geber den Antrieben zuweisen kann, müssen die nachfolgenden, empfohlenen Regeln eingehalten werden.

Empfohlene Regeln:

- Die DRIVE-CLiQ-Leitung von der Control Unit muss folgendermaßen angeschlossen werden:
  - an X200 des ersten Leistungsteils Booksize
  - an X400 des ersten Leistungsteils Chassis
- Die DRIVE-CLiQ-Verbindungen zwischen den Leistungsteilen sind jeweils von der Schnittstelle X201 zu X200 oder entsprechend X401 zu X400 der Folgekomponente anzuschließen.

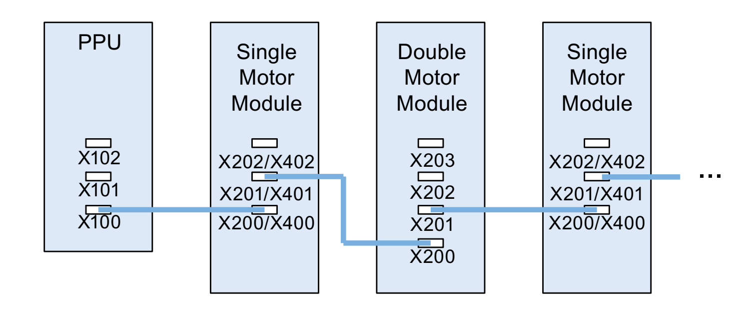


Bild 6-69 Beispiel: DRIVE-CLiQ-Strang

- Der Motorgeber muss an das zugehörige Leistungsteil angeschlossen werden.

Komponente	Anschluss Motorgeber über DRIVE-CLiQ
Single Motor Module Booksize	X202
Double Motor Module Booksize	• Motoranschluss X1: Geber an X202 • Motoranschluss X2: Geber an X203
Single Motor Module Chassis	X402
Power Module Chassis	X402

Hinweis

Wenn ein zusätzlicher Geber an einem Motor Module angeschlossen ist, wird er bei der automatischen Konfiguration diesem Antrieb als Geber 2 zugeordnet. Am Double Motor Module wird ein Geber an X201 dem 2. Vorschub als 2. Messsystem zugeordnet.

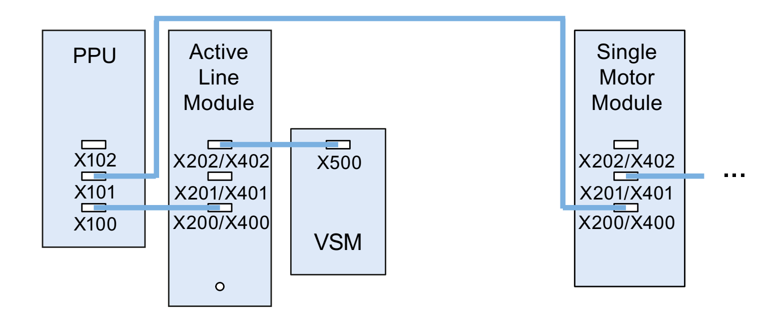


Bild 6-70 Beispiel: Topologie mit VSM bei Booksize- und Chassis-Komponenten

Komponente	Anschluss VSM
Active Line Module Booksize	X202
Active Line Module Chassis	X402
Power Modules	Das VSM wird nicht unterstützt.

6.4.3 Topologieregeln für SMC40

Einsatz des SMC40

Das Sensor Module Cabinet-Mounted SMC40 wird eingesetzt, um Gebersignale von Absolutwertgebern mit EnDat 2.2 auf DRIVE-CLiQ umzusetzen. An das SMC40 sind zwei Gebersysteme mit EnDat 2.2 anschließbar, die unabhängig von einander auf zwei DRIVE-CLiQ-Gebersignale umgesetzt werden.

Anschlussbedingungen

Damit das Sensor Module Cabinet-Mounted SMC40 bei der Erstinbetriebnahme **von der Topologie erkannt wird**, beachten Sie unbedingt die folgenden Regeln:

- Verbinden Sie mindestens eine der DRIVE-CLiQ-Schnittstellen X500/1 oder X500/2 am SMC40 über DRIVE-CLiQ.
- Schließen Sie einen EnDat-Geber an die zugehörige Geberschnittstelle X520/1 (zu X500/1) oder X520/2 (zu X500/2) an.
- Betreiben Sie das SMC40 nur in einer Sterntopologie. Die DRIVE-CLiQ-Buchsen X500/1 und X500/2 können nicht für eine Reihenschaltung verwendet werden.

ACHTUNG
Anzeige im Dialog "Topologie" Das SMC40 wird nur dann in die Ist-Topologie übernommen, wenn die DRIVECLiQ-Schnittstellen X500/x und die zugehörigen Geberschnittstellen X520/x belegt sind. Ohne angeschlossenen Geber wird das SMC40 auch nachträglich nicht in die Topologie eingebunden.

6.5 Klemmenbelegungen

6.5.1 Klemmenbelegung an X122 und X132

X122: Digitale Ein-/Ausgänge der PPU

Pin		Signalname	Status	Bedeutung
1	DI0	DI0	r722.0	Digitaler Eingang 0
2	DI1	DI1	r722.1	Digitaler Eingang 1
3	DI2	DI2	r722.2	Digitaler Eingang 2
4	DI3	DI3	r722.3	Digitaler Eingang 3
5	DI16	DI16	r722.16	Digitaler Eingang 16
6	DI17	DI17	r722.17	Digitaler Eingang 17
7	M2	MEXT2	---	Masse für Pin 1...6
8	P1	P24EXT1	---	+24 V Spannungsversorgung
9	IO8	DI/DO8	r722.8 / r747.8	Digitaler Ein-/Ausgang 8
10	IO9	DI/DO9	r722.9 / r747.9	Digitaler Ein-/Ausgang 9
11	M1	MEXT1	---	Masse für Pin 9, 10, 12, 13
12	IO10	DI/DO10	r722.10 / r747.10	Digitaler Ein-/Ausgang 10
13	IO11	DI/DO11	r722.11 / r747.11	Digitaler Ein-/Ausgang 11
14	M1	MEXT1	---	Masse für Pin 9, 10, 12, 13

X132: Digitale Ein-/Ausgänge der PPU

Pin		Signalname	Status	Bedeutung
1	DI4	DI4	r722.4	Digitaler Eingang 4
2	DI5	DI5	r722.5	Digitaler Eingang 5
3	DI6	DI6	r722.6	Digitaler Eingang 6
4	DI7	DI7	r722.7	Digitaler Eingang 7
5	DI20	DI20	r722.20	Digitaler Eingang 20
6	DI21	DI21	r722.21	Digitaler Eingang 21
7	M2	MEXT2	---	Masse für Pin 1...6
8	P1	P24EXT1	---	+24 V Spannungsversorgung
9	IO12	DI/DO12	r722.12 / r747.12	Digitaler Ein-/Ausgang 12
10	IO13	DI/DO13	r722.13 / r747.12	Digitaler Ein-/Ausgang 13
11	M1	MEXT1	---	Masse für Pin 9, 10, 12, 13
12	IO14	DI/DO14	r722.14 / r747.14	Digitaler Ein-/Ausgang 14

Pin		Signalname	Status	Bedeutung
13	IO15	DI/DO15	r722.15 / r747.15	Digitaler Ein-/Ausgang 15
14	M1	MEXT1	---	Masse für Pin 9, 10, 12, 13

6.5.2 Klemmenbelegung an X242 und X252

X242 Pinbelegung

Pin		Signalname	NC-Variable	Bedeutung
1	nicht verbunden			
2	nicht verbunden			
3	IN1	DIN1	$A_IN[1]	Digitaler NCK-Eingang 1
4	IN2	DIN2	$A_IN[2]	Digitaler NCK-Eingang 2
5	IN3	DIN3	$A_IN[3]	Digitaler NCK-Eingang 3
6	IN4	DIN4	$A_IN[4]	Digitaler NCK-Eingang 4
7	M4	MEXT4		Masse für Pin 3...6
8	P3	P24EXT3		+24 V Spannungsversorgung
9	O1	DOUT1	$A_OUT[1]	Digitaler NCK-Ausgang 1
10	O2	DOUT2	$A_OUT[2]	Digitaler NCK-Ausgang 2
11	M3	MEXT3		Masse für Pin 9, 10, 12, 13
12	O3	DOUT3	$A_OUT[3]	Digitaler NCK-Ausgang 3
13	O4	DOUT4	$A_OUT[4]	Digitaler NCK-Ausgang 4
14	M3	MEXT3		Masse für Pin 9, 10, 12, 13

X252 Pinbelegung

Pin		Signalname	NC-Variable	Bedeutung
1	AO	AOUT		Analog-Ausgang (Spannung für analoge Spindel)
2	AM	AGND		Analog-Masse
3	IN9	DIN9	$A_IN[9]	Digitaler NCK-Eingang 9
4	IN10	DIN10	$A_IN[10]	Digitaler NCK-Eingang 10
5	IN11	DIN11	$A_IN[11]	Digitaler NCK-Eingang 11
6	IN12	DIN12	$A_IN[12]	Digitaler NCK-Eingang 12
7	M4	MEXT4		Masse für Pin 3...6
8	P3	P24EXT3		+24 V Spannungsversorgung
9	O9	DOUT9	$A_OUT[9]	Digitaler NCK-Ausgang 9
10	O10	DOUT10	$A_OUT[10]	Digitaler NCK-Ausgang 10
11	M3	MEXT3		Masse für Pin 9, 10, 12, 13
12	O11	DOUT11	--	• ohne analoge Spindel: Digitaler NCK-Ausgang 11 • mit analoger Spindel: Reglerfreigabe für analoge Spindel (gemäß MD30134 $MA_IS_UNIPOLAR_OUTPUT)
13	O12	DOUT12	--	• ohne analoge Spindel: Digitaler NCK-Ausgang 12 • mit analoger Spindel: Fahrrichtung für analoge Spindel (gemäß MD30134 $MA_IS_UNIPOLAR_OUTPUT)
14	M3	MEXT3		Masse für Pin 9, 10, 12, 13

Siehe auch

Maschinendaten für eine Analog-Spindel (Seite 96)

6.5.3 Beispiel: Verschaltung für eine CU mit Netzschütz

Beispiel

Die abgebildete Verschaltung bezieht sich auf die Belegung der Klemmen in den vorausgegangenen Kapiteln.

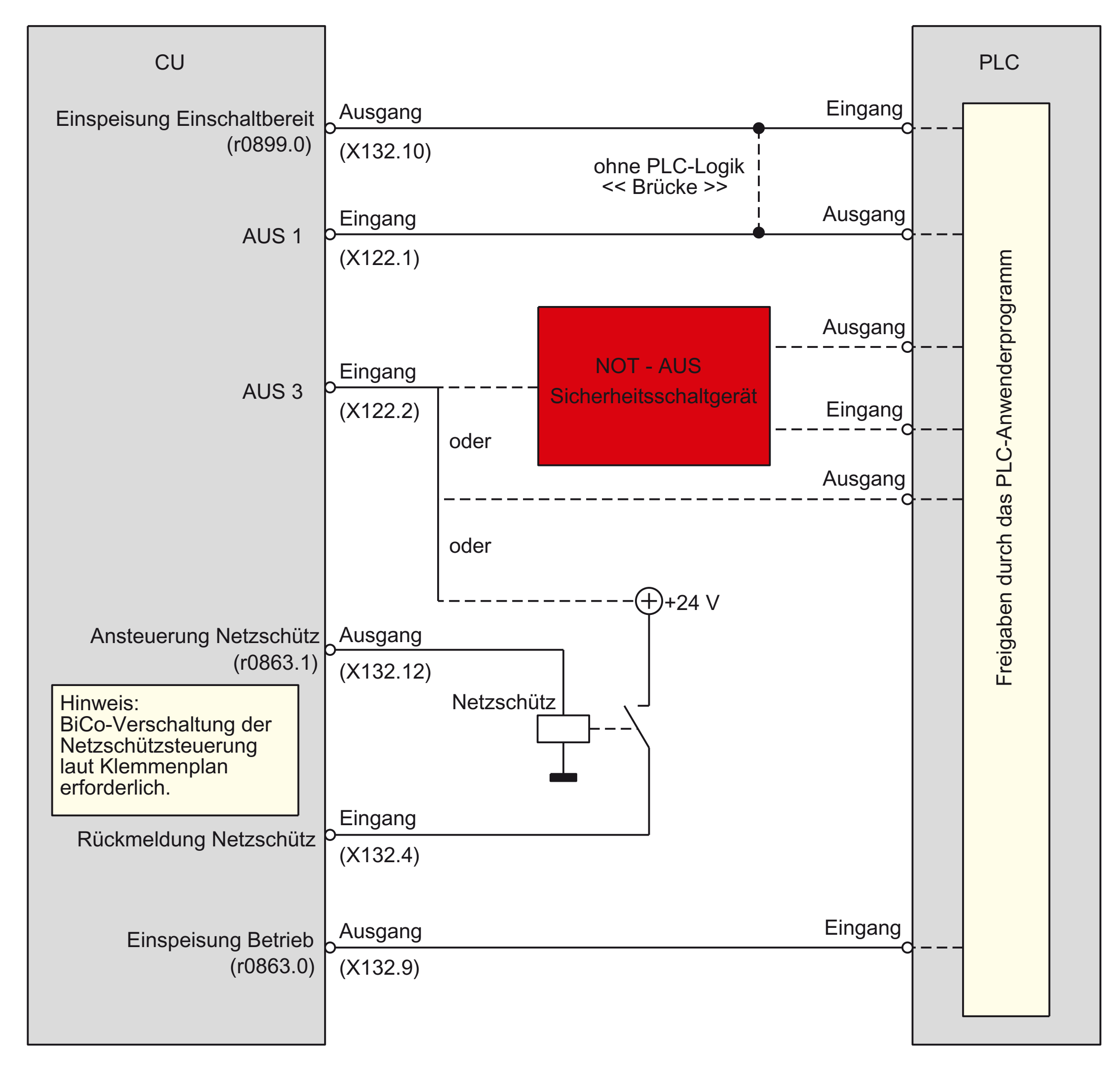
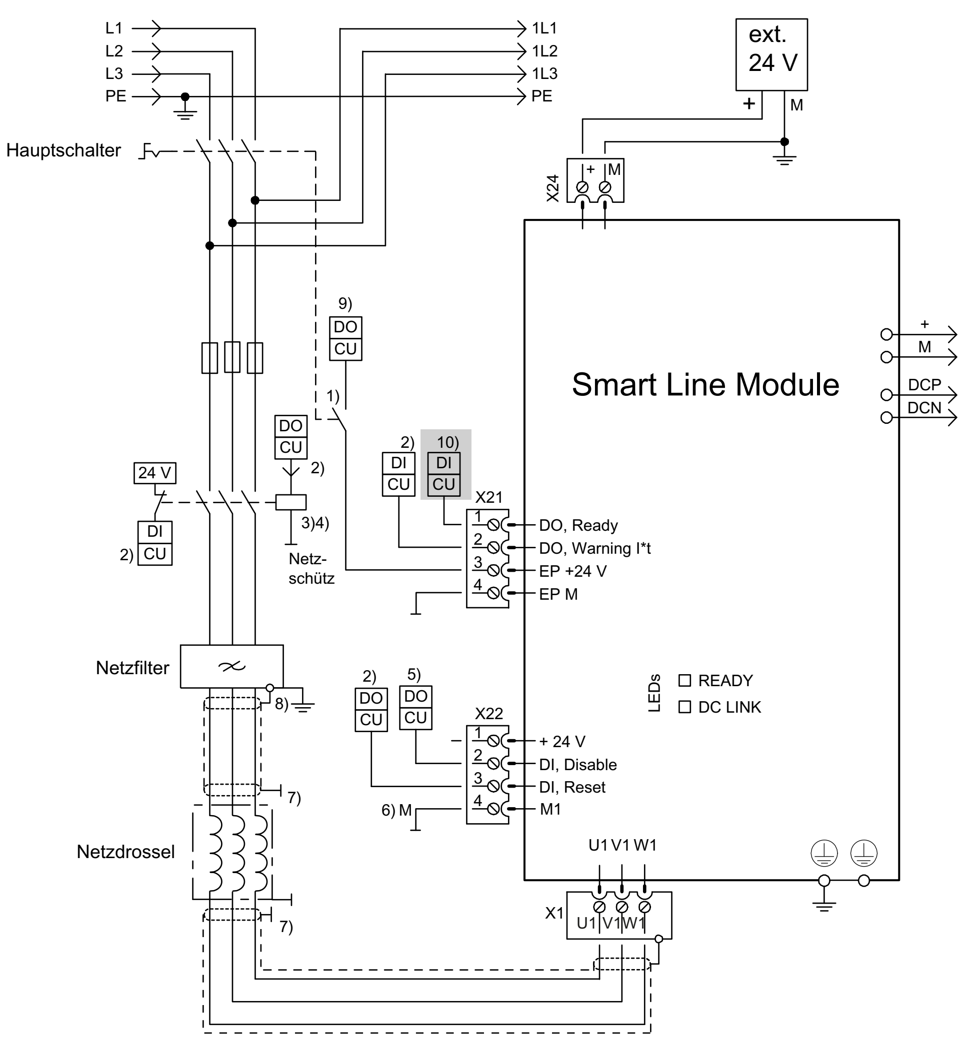


Bild 6-71 Verschaltung einer Control Unit mit Netzschütz

Anschluss Smart Line Module

Die Anschlüsse für die digitalen Ein-/Ausgänge X122 und X132 sind auf der Rückseite der Steuerung.



① Voreilend öffnender Kontakt t >10 ms, zum Betrieb müssen DC 24 V und Masse angelegt werden.

② DI/DO von der Control Unit gesteuert.

③ Kein zusätzlicher Verbraucher hinter dem Netzschütz zugelassen.

④ Die Strombelastbarkeit des DO ist zu beachten, eventuell muss ein Ausgangskoppelglied eingesetzt werden.

⑤ DO high, Rückspeisung deaktiviert (Für dauerhafte Deaktivierung kann eine Brücke zwischen X22 Pin 1 und Pin 2 eingebaut werden.)

⑥ X22 Pin 4 muss mit Masse (extern 24 V) verbunden werden.

⑦ Kontaktierung über Montagerückwand oder Schirmschienen nach EMV-Aufbaurichtlinie

⑧ 5 kW und 10 kW Netzfilter über Schirmanschluss

⑨ Signalausgang der Steuerung, um Rückwirkung der DC 24 V-Versorgung auf EP-Klemme zu vermeiden.

⑩ **Verschalten über BICO auf Parameter p0864 → X122.1**

Bild 6-72 Beispiel: Anschluss SLM

Freigabe EIN/AUS1: Verbindung Smart Line Module Pin X21.1 → X122.1 SINUMERIK 828D

Weitere Ein- und Ausgangssignale mit der PLC-Peripherie verbinden:

- DI → PLC-Eingänge
- DO → PLC-Ausgänge

Siehe auch

Weitere Informationen finden Sie in:

- SINUMERIK 828D Gerätehandbuch PPU
- SINAMICS S120 Gerätehandbuch Leistungsteile Booksize

6.5.4 Anschluss der Messtaster

Messtaster anschließen

Die Messtaster werden sowohl an der SINUMERIK 828D CU als auch an der NX angeschlossen:

1. Messtaster an Klemme X122 Pin 13 / X122 Pin 13 der NX
2. Messtaster an Klemme X132 Pin 13 / X122 Pin 12 der NX

Hinweis

Voraussetzung für das Messen mit der SINUMERIK 828D ist die Parametrierung der dezentralen (lokalen) Messfunktion.

Das zentrale Messen ist mit der SINUMERIK 828D nicht möglich.

Maschinendaten

Folgende Maschinendaten sind zu überprüfen und gegebenenfalls anzupassen:

- **Allgemeine Maschinendaten:**

  MD13200[0] $MN_MEAS_PROBE_LOW_ACTIVE = 0 oder 1

  MD13200[1] $MN_MEAS_PROBE_LOW_ACTIVE = 0 oder 1

  Wert 0 = ausgelenkter Zustand 24 V   (Voreinstellung)

  Wert 1 = ausgelenkter Zustand 0 V

  MD13210 $MN_MEAS_TYPE = 1   Dezentrales Messen

- **Achsspezifische Maschinendaten:**

  MD30244[0] $MA_ENC_MEAS_TYPE = 1   für alle Achsen

  MD30244[1] $MA_ENC_MEAS_TYPE = 1   für alle Achsen

Hinweis

Die Maschinendaten MD13210 und MD30244 sind auf den Wert 1 voreingestellt und können nicht geändert werden! (Datenklasse: SYSTEM)

- **Control Unit Parameter:**

  p0680[0] Zentraler Messtaster Eingangsklemme = 0

  p0680[1] Zentraler Messtaster Eingangsklemme = 0

  p0680[2] Zentraler Messtaster Eingangsklemme = 0

  Der 1. Messtaster wird an Klemme X122 Pin 13, der 2. Messtaster an Klemme X132 Pin 13 der SINUMERIK 828D und - wenn vorhanden - auf der NX10 Baugruppe an Klemme X122 Pin 12 angeschlossen.

  Voraussetzung ist die Umschaltung des Pins X132.13 von Ausgang auf Eingang.

  CU Eingang oder Ausgang einstellen = p0728 Bit 15 auf 0 setzen (DI/DO X132.13)

- **Antriebsparameter:**

  p0488[0] Messtaster 1 Eingangsklemme: Geber 1 = 3 → Stecker X 122.13

  p0488[1] Messtaster 1 Eingangsklemme: Geber 2 = 3 → Stecker X 122.13

  p0488[2] Messtaster 1 Eingangsklemme: Geber 3 = 0 → nicht verwendet

  p0489[0] Messtaster 2 Eingangsklemme: Geber 1 = 6 → Stecker X 132.13

  p0489[1] Messtaster 2 Eingangsklemme: Geber 2 = 6 → Stecker X 132.13

  p0489[2] Messtaster 2 Eingangsklemme: Geber 3 = 0 → nicht verwendet

  Für alle Achsen, die auf der NX-Baugruppe parametriert sind, muss der 2. Messtaster an Klemme X122.12 parametriert werden:

  p0489[0] Messtaster 2 Eingangsklemme: Geber 1 = 6 → Stecker X 122.12

  p0489[1] Messtaster 2 Eingangsklemme: Geber 2 = 6 → Stecker X 122.12

  p0489[2] Messtaster 2 Eingangsklemme: Geber 3 = 0 → nicht verwendet

---

Hinweis

Es sind alle Antriebe zu parametrieren.

Messtaster - Status

DB2700	Allgemeine Signale von NCK [r] Nahtstelle NCK → PLC							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0	--	--	--	--	--	--	NOT AUS aktiv	---
DBB1	System inch-Maß	--	--	--	--	--	Messtaster betätigt	
							Messtaster 2	Messtaster 1

Antrieb optimieren 7

7.1 Einleitung

Geänderte Maschinendaten aktivieren

Bei diesem Verfahren werden bestimmte Änderungen von Maschinendaten aktiviert. Auf der rechten Seite des Parameterwerts wird angezeigt, wie die jeweilige Änderung aktiviert wird:

po	Die geänderten Maschinendaten erfordern einen "NCK Reset", um wirksam zu werden.
re	Die geänderten Maschinendaten erfordern eine Aktivierung durch die Taste <RESET>.
cf	Die geänderten Maschinendaten erfordern eine Aktivierung durch den Softkey "MD wirksam setzen".
so	Die geänderten Maschinendaten sind sofort aktiv.

Antriebsdaten sichern

Wenn der Antrieb eingeschalten oder ein "Drive Reset" durchgeführt wird, werden die Daten des SINAMICS S120 aus der gespeicherten Datei in den entsprechenden Antrieb geladen. Um Änderungen in den Antriebsdaten nach dem nächsten Aus-/Einschalten oder nach einem "Drive Reset" dauerhaft zu übernehmen, müssen die Daten gespeichert werden.

Hinweis

Ein "NCK Reset" hat keinen Einfluss auf die Antriebsdaten.

Vorgehensweise:

1. Wählen Sie den Softkey "Antriebsparameter" und dann den Softkey "Speichern/Reset".
2. Wählen Sie den Softkey "Speichern" und zuletzt den Softkey "Antriebssystem".

Meldungen im linken unteren Bereich des Bildschirms zeigen den Status des Speichervorgangs an. Wenn der Speichervorgang abgeschlossen ist, wird folgende Meldung angezeigt: "Alle Antriebsobjekte gesichert".

Steuerungsblockdiagramm

Die Steuerung und der Antrieb enthalten drei kaskadierte geschlossene Regelkreise:

- Stromregelkreis
- Drehzahlregelkreis
- Lageregelkreis

Die einzelnen Regelkreise werden jeweils aufeinander folgend optimiert, beginnend mit dem Stromregler, dann der Drehzahlregler und zuletzt der Lageregler. Die dynamischen Maschinengrenzen für Geschwindigkeit, Beschleunigung und Ruck werden im Interpolator eingestellt. Der Drehzahlregelkreis wird immer über den Motorgeber geschlossen, während der Lageregelkreis über den Motorgeber oder über das direkte Messsystem geschlossen werden kann.

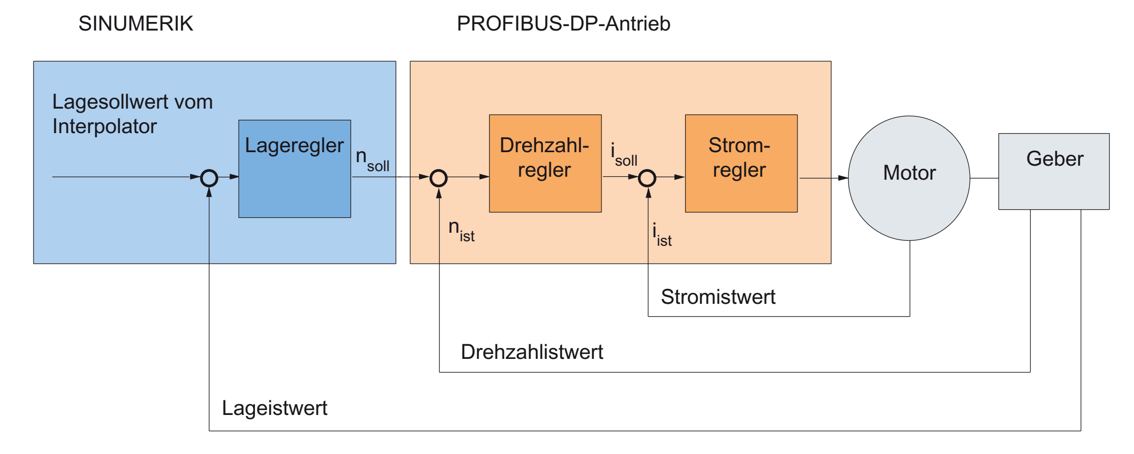


Bild 7-1 Steuerungsstruktur

Siehe auch

In den folgenden Kapiteln werden die einzelnen Schritte zur Optimierung der Regelkreise mit SINUMERIK Operate beschrieben:

- Automatische Servo-Optimierung (Seite 183)
- Achsruck (Seite 196)
- Drehmomentauslastung (Seite 204)
- Kreisformtest (Seite 208)
- Spindeloptimierung (Seite 217)

Literatur

Weitere Informationen finden Sie in folgenden Handbüchern:

- Inbetriebnahmehandbuch SINUMERIK Operate (IM9); Kapitel "Trace"
- Funktionshandbuch Grundfunktionen; Achsüberwachungen, Schutzbereiche (A3)
- Funktionshandbuch Grundfunktionen; Geschwindigkeiten, Soll-/Istwertsysteme, Regelung (G2)
- Funktionshandbuch Erweiterungsfunktionen; Kompensationen (K3)

7.2 Automatische Servo Optimierung

7.2.1 Zielsetzung

Einleitung

Die Funktion "Automatische Servo Optimierung" verwenden Sie, um den Strom-, Drehzahl- und Lageregelkreis mit Vorsteuerung zu optimieren. Ziel ist die Feineinstellung einer Achse oder Spindel unter Verwendung von SINUMERIK Operate.

- Durchführung von Frequenzgangmessungen
- Berechnung des Drehzahlregelkreises
- Einstellung von Stromsollwertfiltern
- Berechnung des Lageregelkreises
- Überprüfung des tatsächlichen Verhaltens der geplanten Regelkreise
- Vorsteuerungsmodus: äquivalente Drezahlregelzeit
- Interpolationspfade

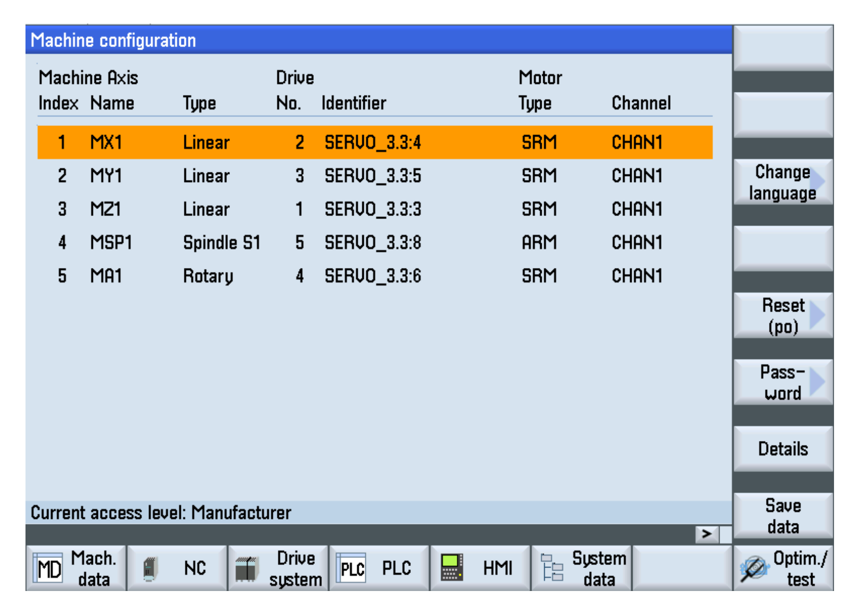
Vor der automatischen Optimierung wird die dynamische Steifigkeitsregelung in MD32640[0] $MA_STIFFNESS_CONTROL_ENABLE = 1 gesetzt. Damit ist eine Lageregelung mit einer schnelleren Zykluszeit möglich. Bei Verwendung der dynamischen Steifigkeitsregelung ist MD32110 $MA_ENC_FEEDBACK_POL = 1 zu setzen. Wenn eine Umkehrung der Polarität notwendig ist, muss im Antrieb der Parameter p0410 "Geber Invertierung Istwert" eingestellt werden:

- p0410[0]=1 Drehzahlistwert invertieren
- p0410[1]=1 Lageistwert invertieren

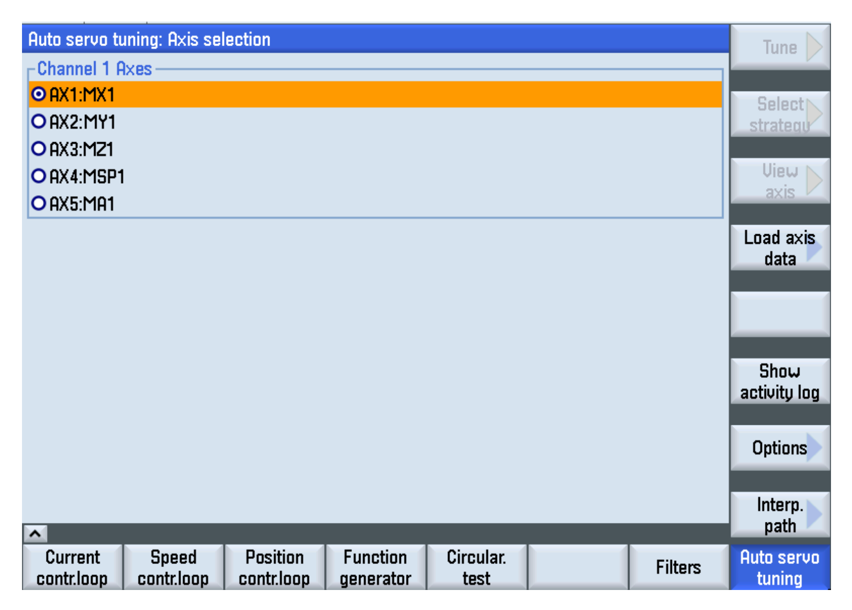
Wird die Funktion "Automatische Servo Optimierung" bei einer Hauptspindel angewendet, kann es sein, dass kein PLC-Signal für die Servo-Freigabe vorhanden ist. Der Maschinenhersteller sollte eine Methode zur Verfügung stellen, dieses PLC-Signal für die Optimierung freizuschalten: zum Beispiel eine spezielle Tastenkombination oder das Setzen eines Merkers im PLC-Status.

Automatische Servo Optimierung

Die Funktion "Automatische Servo Optimierung" starten Sie im Bedienbereich "Inbetriebnahme" → Softkey "Optimierung/Test".



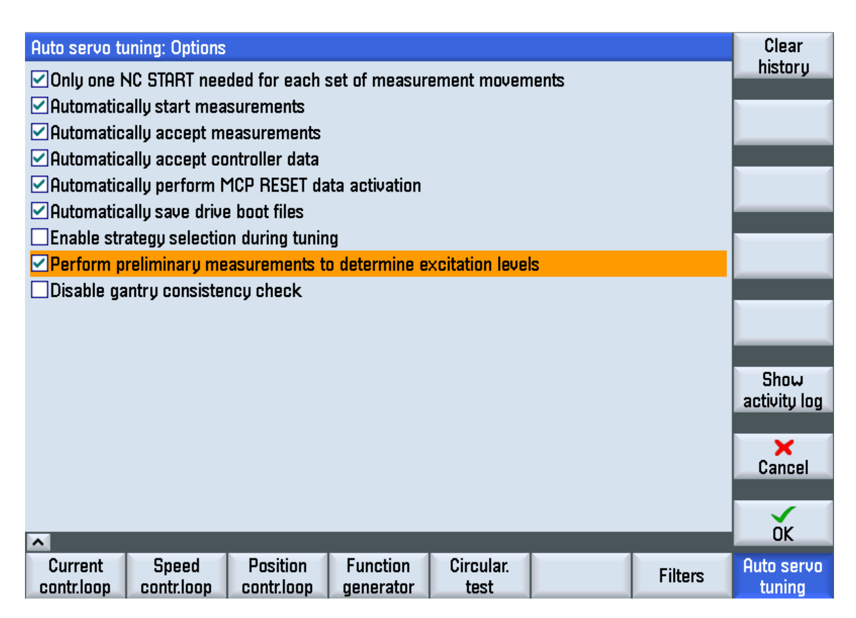
Markieren Sie die Achse mit Hilfe der Cursor-Tasten und wählen Sie mit der Taste <SELECT> z. B. die Achse X1 aus:



Der Dialog zeigt an, ob eine Achse bereits optimiert ist. Eine optimierte Achse ist mit Uhrzeit und Datum der Optimierung gekennzeichnet.

Optionen prüfen

Überprüfen Sie, ob alle Optionen für einen automatischen Ablauf gesetzt sind: Softkey "Optionen". Damit ist eine Optimierung durch Aktivieren eines einzigen Softkeys möglich. Folgen Sie den Meldungen und Anweisungen am Bildschirm.



Die Option "Strategieauswahl während der Optimierung erlauben" wird nur für Experten empfohlen.

7.2.2 Optimierungsstrategie einstellen

Optimierungsstrategie

Für die Achs-Strategien, Drehzahlregler- und Lagereglerstrategien wird empfohlen, die Voreinstellung (Strategien 102, 303 und 203, siehe Abbildung) zu verwenden:

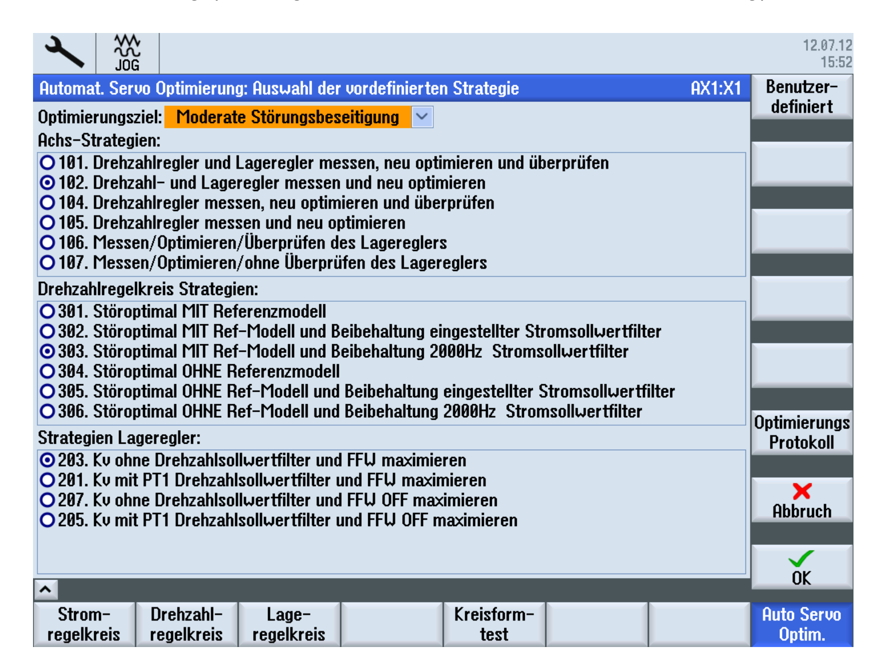


Bild 7-2 Auswahl der vordefinierten Strategie

Als Optimierungsziel voreingestellt ist "Moderate Störungsbeseitigung". Die Auswahl hängt von den mechanischen Eigenschaften der Maschine/Achse ab. Die Option "Maximale Störungsbeseitigung" sollte nicht für eine "schwache" Achse verwendet werden.

Optimierungsziel

Beschreibung des Optimierungsziels:

- **Auswahl: "Maximale Störungsbeseitigung"**

  Die Drehzahl- und Lagereglerverstärkung (Kv-Faktor) wird mit Maximalwerten und minimaler Robustheit optimiert.

  – Anwendung: Hochgeschwindigkeitsbearbeitung bei maximaler Unterdrückung aller Störkräfte wie beispielsweise Reibung, Zähne des Antriebsriemens, starke Schneidkräfte wie z. B. bei der Bearbeitung von Titan.

    Empfohlen für Hochgeschwindigkeitsbearbeitung mit Linearmotoren.

  – Voraussetzung: Die Maschine muss einen starren Aufbau aufweisen; die bewegten Massen verändern sich nicht wesentlich.

- **Auswahl: "Moderate Störungsbeseitigung"**

  Der Drehzahl- und Lageregler wird mit 80% der Maximalverstärkung und einer guten Robustheit optimiert.

  – Anwendung: Es gibt viele Verwendungsarten.

  – Voraussetzung: Keine hohen Erwartungen an den Maschinenaufbau; empfohlen für allgemeine Verwendungszwecke.

- **Auswahl: "Dämpfung optimal"**

  Der Drehzahlregler wird so optimiert, dass er eine maximale Dämpfung erreicht, um Schwingungen zu verhindern und einen guten Lageregelungswert zu erreichen.

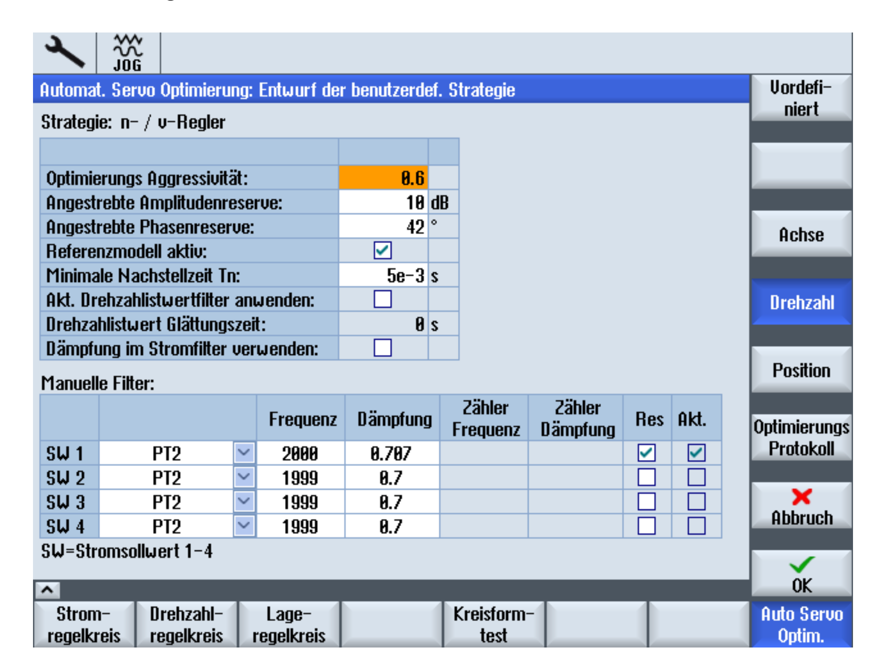
  – Anwendung: Große Maschinen mit großen bewegten Massen.

  – Voraussetzung: Aufgrund der großen Massen/Trägheit ein schwacher Maschinenaufbau. Die Größe der Maschine begründet eine geringe Dynamik. Empfohlen beispielsweise für Fahrständer, bei denen die Bedienerkabine mit dem Ständer läuft. Kann verwendet werden, wenn die "Dämpfung optimal"-Prüfung eine sehr geringe Lagereglerverstärkung ergibt.

Benutzerdefinierte Einstellungen

Über den Softkey "Benutzerdefiniert" wird der Optimierungsvorgang neu definiert.

Drehzahlregler einstellen:



Die wichtigen Einstellungen lauten: Optimierungs-Aggressivität und Minimale Nachstellzeit Tn.

- Optimierungs-Aggressivität:

  Dieser Parameter bestimmt die Einstellung von Kp und Tn basierend auf Stabilitätsgrenzen.

  – Voreinstellung = 0,6

  – Min = 0 [maximal stabil]

  – Max = 1 [maximal aggressiv]

- Minimale Nachstellzeit Tn:

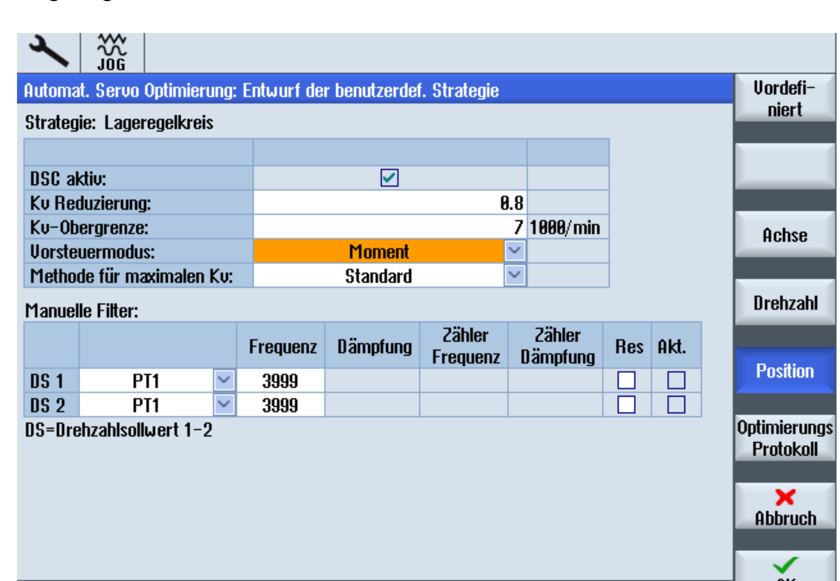
  Dieser Parameter verhindert, dass die Autom. Servo Optimierung die Integratorzeit des Drehzahlregelkreises zu niedrig einstellt. Hätte Autom. Servo Optimierung sonst einen niedrigeren Wert als diesen Parameter eingestellt, wird der tatsächlich verwendete Wert auf den durch die Minimale Nachstellzeit Tn eingestellten Wert begrenzt.

  – Voreinstellung = 5,0 ms

  – Min = 0,5 ms

  – Max = 100 ms

Lageregler einstellen:



Hier lautet die wichtige Einstellung: Kv (Lageregler) Reduzierung

- Kv Reduzierung:

  Dieser Parameter verhält sich wie die Aggressivität des Lagereglers.

  Er verursacht eine bestimmte Reduzierung des von Autom. Servo Optimierung berechneten maximalen Kv.

  Der maximale Kv ist der größte Kv, der zu einem Nullpositionsüberschwingen bei deaktiviertem Vorschub führen würde.

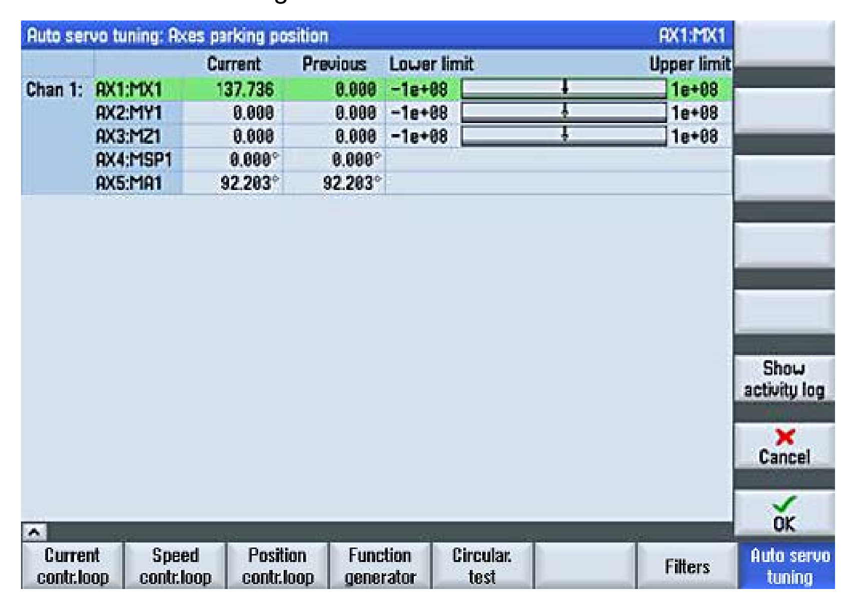
  – Voreinstellung = 0,8
  – Min = 0,1
  – Max = 1 [keine Reduzierung]

7.2.3 So starten Sie die Automatische Servo Optimierung

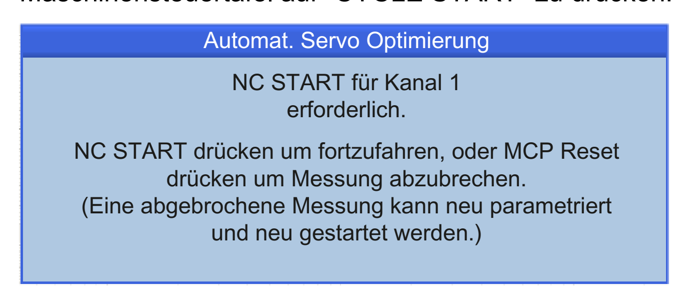
Automatische Servo Optimierung starten

Vorgehensweise:

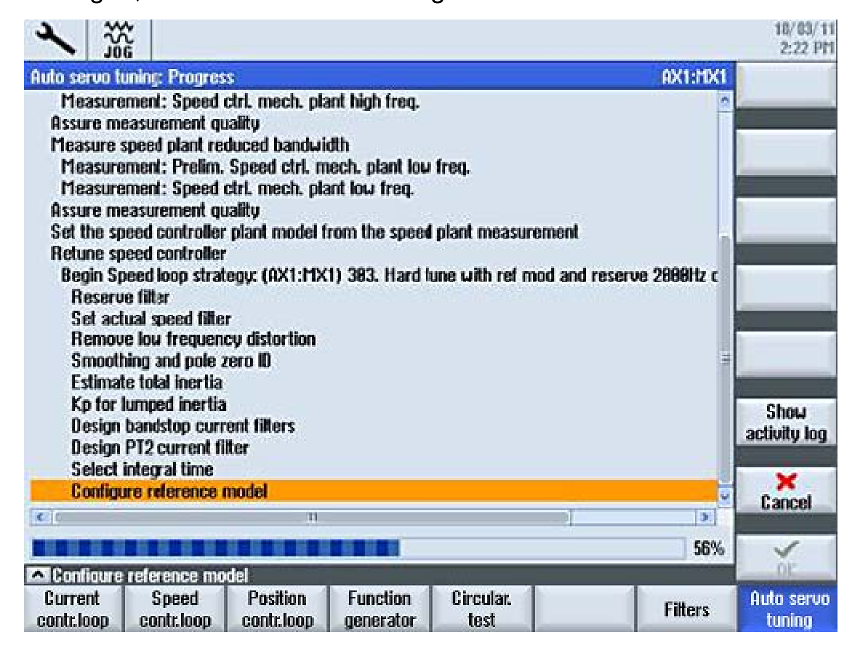
1. Stellen Sie sicher, dass sich die Achse, die optimiert werden soll, in einer sicheren Position befindet. Ist dies nicht der Fall, verfahren Sie die Achse im JOG-Betrieb n die Mitte ihres Verfahrweges:



2. Zur Bestätigung drücken Sie "OK". Danach kommt die Aufforderung auf der Maschinensteuertafel auf "CYCLE START" zu drücken:

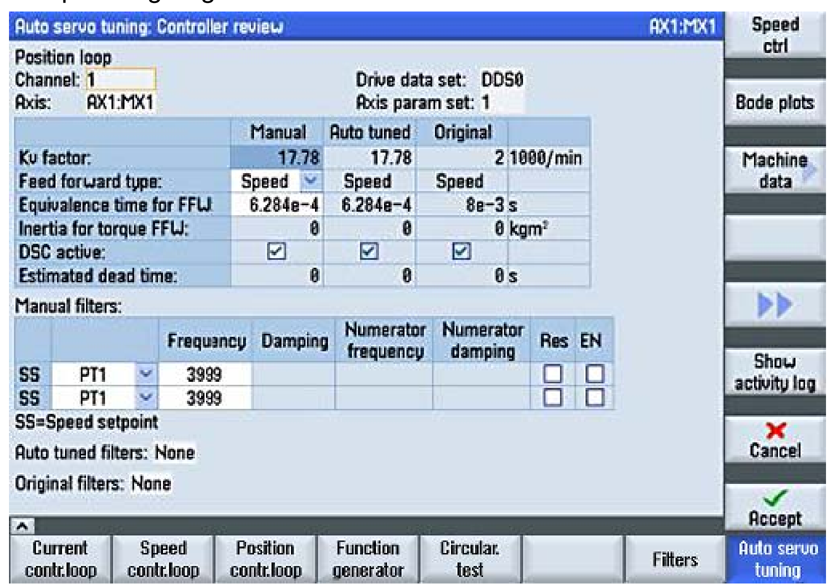


3. Im weiteren Verlauf des automatischen Optimierungsverfahrens werden weitere Anzeigen am Bildschirm ausgegeben. Die Taste "CYCLE START" ist mehrmals zu betätigen, um die nächste Messung zu starten:

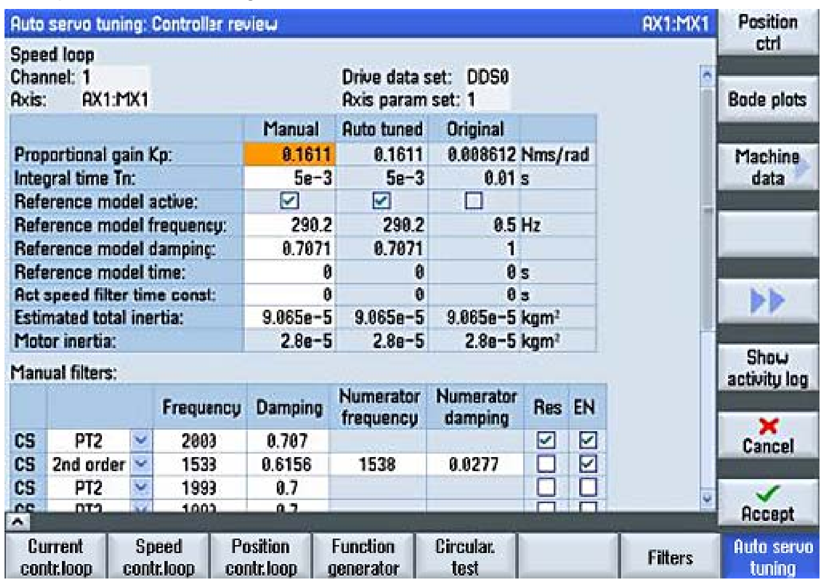


4. Nach Abschluss der Optimierung erscheint die folgende Anzeige, in dem die neuen Werte und die Originalwerte überprüft werden.


Beispiel: Lageregler

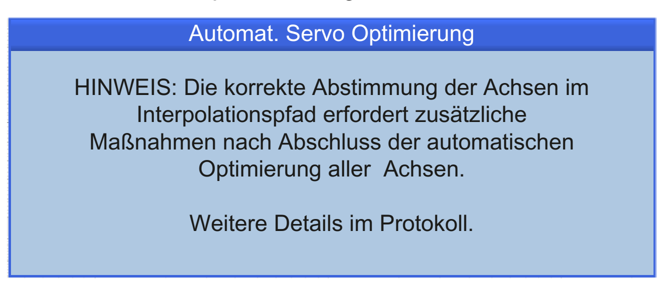



Beispiel: Drehzahlregler



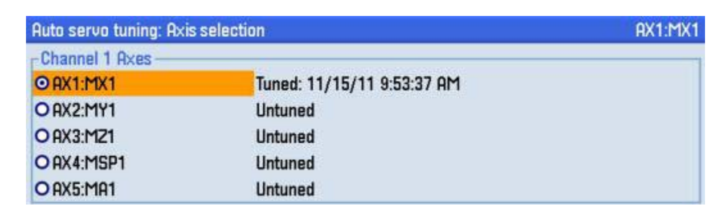
Die neuen Werte werden übernommen, wenn der Softkey "Übernehmen" gedrückt wird.

5. Die folgende Meldung weist darauf hin, auch die anderen Achsen zu optimieren, um ein ordentliches Interpolationsergebnis zu erhalten:

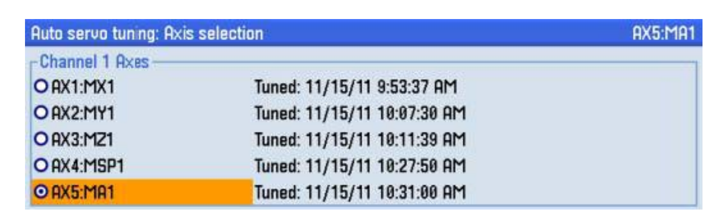


Ergebnis der Optimierung

Anschließend werden Uhrzeit und Datum der Optimierung ausgegeben:



Auf dieselbe Weise werden auch die anderen Achsen optimiert:



7.2.4 So optimieren Sie die Interpolationpfade

Einleitung

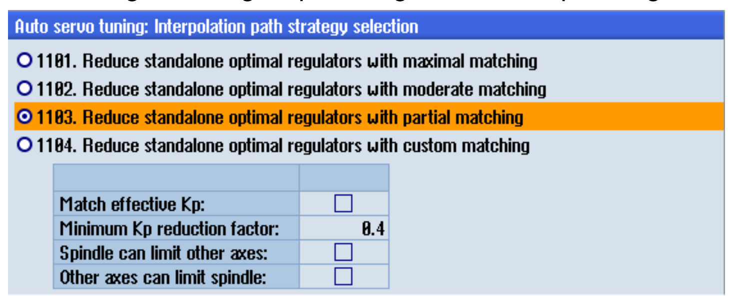
Interpolation bedeutet, dass die Achsen, die die Kontur ausführen, denselben Schleppfehler aufweisen müssen. Nach der Optimierung der Einzelachsen besitzen die Achsen unterschiedliche Steuerungseinstellungen, die nur für die jeweilige Achse optimal sind. Um einen einheitlichen Schleppfehler zu gewährleisten, müssen die Steuerungseinstellungen so angepasst werden, dass sie der Achse mit der langsamsten Reaktion entsprechen. Dies schließt den Kv-Faktor und den Vorsteuermodus ein.

Strategieauswahl für die Interpolation

Vorgehensweise:

1. Legen Sie die Strategie für den Interpolationspfad fest. Es wird die Voreinstellung verwendet:

   • 1103 "Eigenständige Optimalregler mit Teilanpassung reduzieren".

   Bedeutung der Strategien:

   – Maximalanpassung: Die wirksame Verstärkung Kp ist angepasst.
     Die Proportionalverstärkung Kp kann bis auf 10% des ursprünglichen Wertes reduziert werden.
     Die Verstärkung Kp der Vorschubachse wird nicht reduziert, um den niedrigeren wirksamen Wert der Hauptspindel zu nutzen. Die Verstärkung Kp der Hauptspindel wird nicht reduziert, um die niedrigere wirksame Verstärung Kp der Vorschubachse zu nutzen.

   – Moderate Anpassung: Die wirksame Verstärkung Kp ist angepasst.
     Die Proportionalverstärkung Kp kann bis auf 50% des ursprünglichen Wertes reduziert werden.
     Die Verstärkung Kp der Vorschubachse wird nicht reduziert, um den niedrigeren wirksamen Wert der Hauptspindel zu nutzen.
     Die Verstärkung Kp der Hauptspindel wird nicht reduziert, um die niedrigere wirksame Verstärung Kp der Vorschubachse zu nutzen.

   – Teilanpassung:
     Die wirksame Verstärkung Kp ist nicht angepasst.

   – Benutzerdefinierte Anpassung:
     Die Strategieauswahl erfolgt durch benutzerdefinierte Einstellungen. Die Kontrollkästchen sind aktivierbar.

Allgemeine Einstellungen

Einstellungen für "Pfad optimal":

- Dynamische Steifigkeitsregelung (DSC)

  MD32640 $MA_STIFFNESS_CONTROL_ENABLE ist für alle Achsen entweder aktiviert oder deaktiviert. Wenn DSC aktiviert ist, sind höhere K_V-Faktoren möglich.
- Modus Vorsteuerung
  - Wenn alle Achsen mit Momentenvorsteuerung optimiert sind, dann bleibt diese Einstellung für den Interpolationspfad bestehen.
  - Wenn die Vorsteuerung für die Achsen unterschiedlich auf 3 oder 4 eingestellt ist, dann wird für alle Achsen MD32620 $MA_FFW_MODE = 3 gesetzt.
  - Wenn eine Achse ohne Vorsteuerung optimiert worden ist, wird für alle Achsen MD32620 $MA_FFW_MODE = 0 gesetzt.

Einstellungen für "Achse optimal":

- Der K_V-Verstärkungsfaktor wird berechnet; die niedrigste Einstellung wird für alle Achsen übernommen: K_V-Faktor wird immer angepasst!
- Proportionalverstärkung Kp/Trägheitsmoment für spätere Anpassung der Achsen.
- Die Nachstellzeit Tn wird für alle Achsen angepasst.
- Das Referenzmodell (Drehzahlregler) ist aktiviert, d.h. es wird angepasst.
- Ersatzzeitkonstante Drehzahlregelkreis Vorsteuerung wird berechnet, die höchste Einstellung wird für alle Achsen übernommen.

MD32402 $MA_AX_JERK_MODE = 0 Filtertyp für axiale Ruckbegrenzung ist nicht aktiviert.

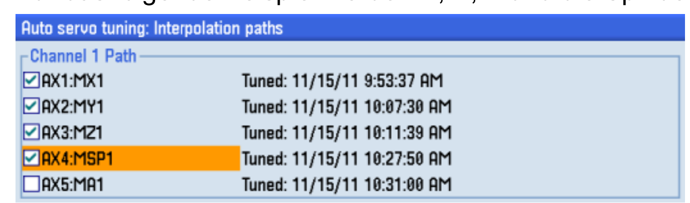
Interpolationspfad

Im nächsten Schritt werden die Achsen für die Optimierung der Interpolationspfade ausgewählt.

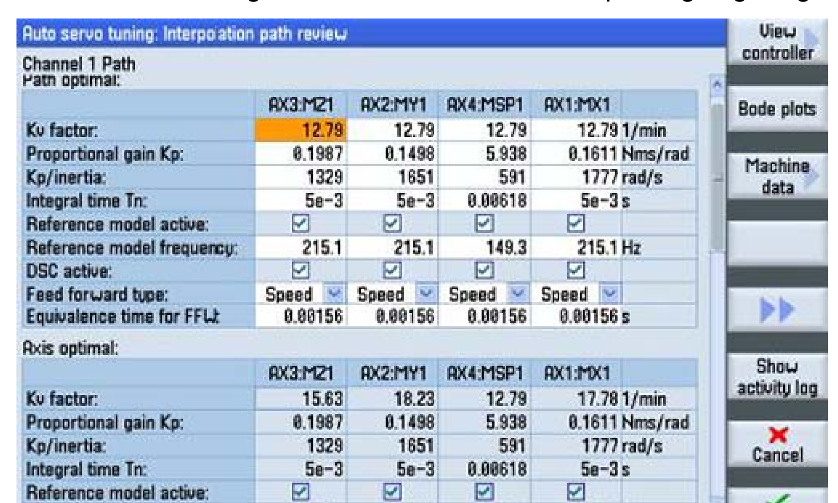
Vorgehensweise:

1. Drücken Sie den Softkey "Interpolationspfade". Die Achsen, die zusammen interpolieren, werden "Kanal 1" zugewiesen.
2. Drücken Sie den Softkey "Editieren & optimieren".
3. Wählen Sie nur die Achsen aus, die zusammen interpolieren können.

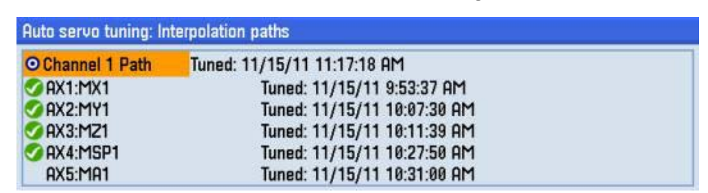
   Für das folgende Beispiel werden X, Y, Z und die Spindel ausgewählt:



4. Die neue Einstellung wird berechnet und zur Überprüfung angezeigt:



5. Wählen Sie "Übernehmen" und bestätigen Sie mit "OK".



Die automatische Optimierung ist damit abgeschlossen.

7.3 Achsruck optimieren

7.3.1 Achsruck überprüfen

Ruckbegrenzung

Im nächsten Schritt wird die ruckbegrenzte Beschleunigung optimiert: Ruck wird verwendet, um das Beschleunigungs- und Ansprechverhalten durch Anlegen einer Zeit für die Änderung der Beschleunigung zu glätten.

Der Ruck wird in Einheiten/s^3 definiert, d.h. Beschleunigung [Einheiten/s^2] / Zeit [s].

Die Zeit, während der die Beschleunigungsveränderung stattfindet, ist definiert gemäß: t [s] = Beschleunigung [Einheiten/s^2] / Ruck [Einheiten/s^3].

Achsen müssen nicht den gleichen Ruckwert haben. Das Ergebnis einer ruckbegrenzten Beschleunigung sind abgerundete Ecken am Anfang und am Ende der Beschleunigungsphase des Geschwindigkeitsprofils und geglättete Bewegungen.

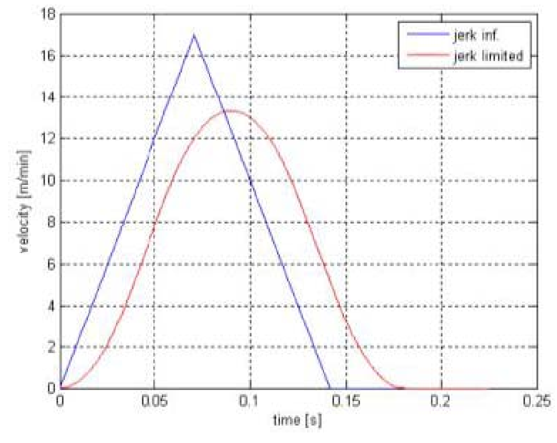
- Ruckbegrenzung ist aktiv, wenn "SOFT" programmiert ist.
- Ruckbegrenzung ist deaktiviert, wenn "BRISK" programmiert ist.

Maschinendaten für Ruck

Es wird empfohlen, die Voreinstellung MD20150[20] = 2 d.h. "SOFT" zu verwenden:

- MD20150[20] = 1 dann ist "BRISK" im Einschaltzustand aktiv (Beschleunigung ohne Ruckbegrenzung).
- MD20150[20] = 2 dann ist "SOFT" im Einschaltzustand aktiv (Beschleunigung mit Ruckbegrenzung).

Beispiel:



blau	Beschleunigung ohne Ruckbegrenzung
rot	Beschleunigung mit Ruckbegrenzung

Positionierverhalten der Achsen

Das Positionierverhalten der Achsen kann mit der Trace-Funktion durch Sollpositionsschritte über ein Teileprogramm mit verschiedenen Vorschüben überprüft werden.

- Wird ein direkter Maßstab verwendet, dann ist das Ansprechverhalten des Schlittens leicht überschaubar.
- Wird ein Motorgeber verwendet, um den Lageregelkreis zu schließen (indirekte Rückkopplung), dann ist das Ansprechverhalten des Schlittens für den geschlossenen Lageregelkreis mit der Trace-Funktion nicht erkennbar.

Bei Maschinen mit SINUMERIK 828D werden allgemeinen Ruckwerte zwischen 20 und 100 verwendet. Das Testprogramm sollte die Achse über einen ausreichenden Abstand verfahren, so dass die programmierte Vorschubgeschwindigkeit erreicht werden kann.

Eine Verweilzeit von z. B. 0,5 Sekunden sollte, nachdem die Vorschubgeschwindigkeit und die Soll-Position erreicht ist, verwendet werden. Die erste programmierte Vorschubgeschwindigkeit sollte 50% des maximalen Achsvorschubs betragen. Dieses Verhalten sollte bei allen Inkrementen des Vorschub-Overrides, bis zur maximalen Vorschubgeschwindigkeit überprüft werden.

7.3.2 Teileprogramm für Achsruck

Teileprogramm zum Überprüfen

Das folgende NC-Programm kann im Automatik- oder MDA-Betrieb verwendet werden, abhängig von der Konfiguration der verwendeten Maschine. Das Programm enthält Befehle für das Aktivieren/Deaktivieren des Trace-Triggers:

```
FFWON

SOFT

$AN_SLTRACE=0; Reset Start Servo-Trace-Trigger

LAB:

G01 Y210 F10000

G04 F0.5

$AN_SLTRACE=1; Start Servo-Trace-Trigger

Y260

G04 F0.5

GOTOB LAB

M30
```

Einsehen der Trace-Ergebnisse:

Wählen Sie <MENU SELECT> → Bedienbereich "Diagnose", Menüfortschalttaste → Softkey "Trace" und drücken Sie <CYCLE START>:

Die Achse bewegt sich und die ausgewählten Signale werden für 5 Sekunden überwacht. Anschließend wird das Ergebnis angezeigt.

Maschinendaten:

Um das Positionierverhalten ohne den Einfluss des Lagereglers zu prüfen, sollte die Lagereglerverstärkung K_v-Faktor = 0 gesetzt werden. Stellen Sie Positioniertoleranzen ein, um während der Prüfung Fehler zu vermeiden:

- MD32200 $MA_POSCTRL_GAIN = 0
- MD36012 $MA_STOP_LIMIT_FACTOR = 100
- MD36400 $MA_CONTOUR_TOL = 20

Trace initialisieren

Der Trace wird über den Softkey "Trace starten" gestartet und initialisiert. Dies hängt von der Auswahl im Dialog "Einstellungen" ab. Indiesem Beispiel wird der Trace über einen Befehl des Teileprogramms gestartet. Drücken Sie den Softkey "Trace starten", um den Trace zu initialisieren:

Der Trace ist jetzt initialisiert und wird gestartet, sobald über das Teileprogramm der Trigger eingestellt ist.

7.3.3 Trace-Signale auswählen

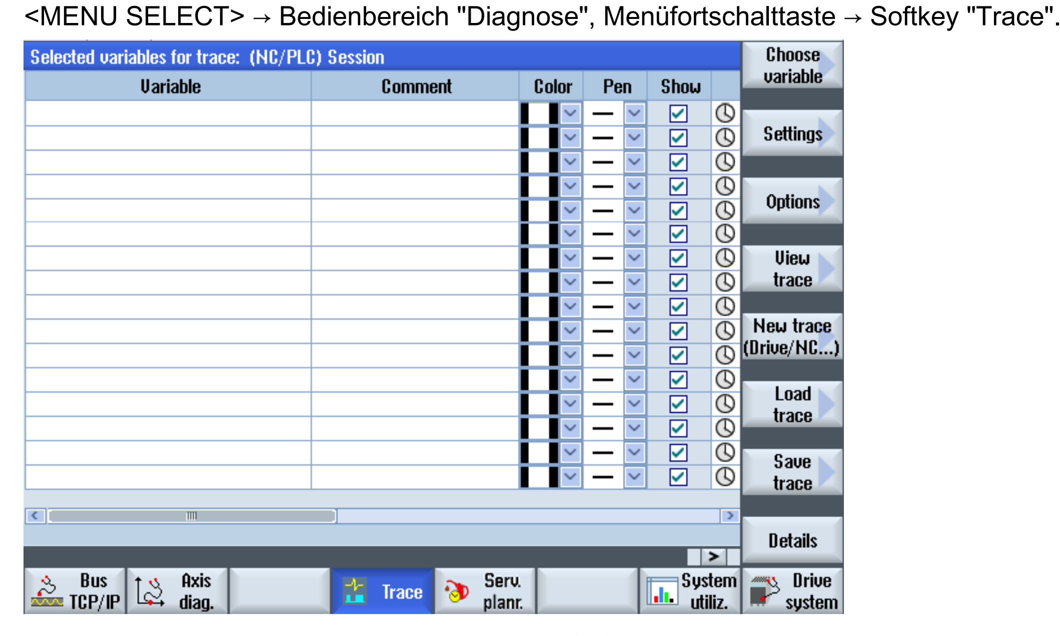
Trace-Signale auswählen

Es wird ein Trace der Lagesollwerte und Lageistwerte benötigt.

Vorgehensweise:

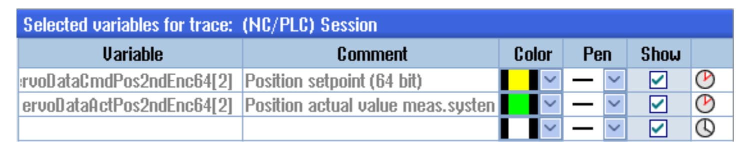
1. Verwenden Sie die folgende Tastenkombination, um die Funktion Trace aufzurufen:

   <MENU SELECT> → Bedienbereich "Diagnose", Menüfortschalttaste → Softkey "Trace".



2. Die benötigten Variablen werden ausgewählt: Softkey "Variable wählen".

3. Verwenden Sie den Softkey "Filter/Suchen", um die Liste der Variablen einzugrenzen. Folgende Filter stehen dafür zur Verfügung:
   - Systemvariablen
   - NC
   - Achse
   - Kanal
   - Betriebsartengruppen (BAG)
   - Servo
   - Globale Anwenderdaten (GUD)
   - PLC
   - PLC Alarme
   - Alle (kein Filter)
4. Setzen Sie den Filter auf "Servo", und wählen Sie die beiden Variablen Lagesollwert und Lageistwert z. B. für die Y-Achse aus:



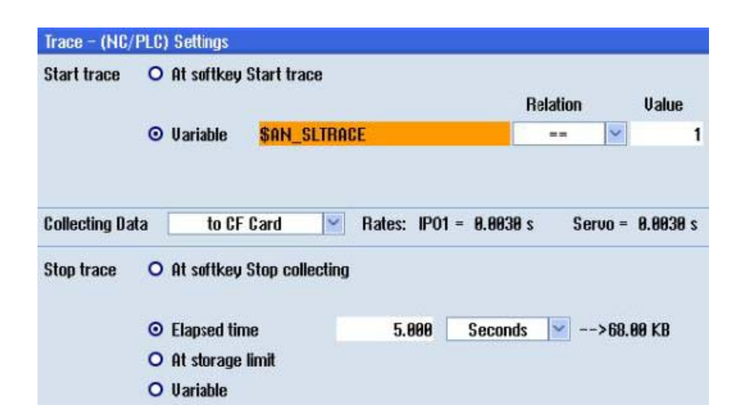
Dauer und Startbedingungen des Trace

Der Trace kann manuell oder automatisch gestartet werden. Außerdem kann die Dauer des Trace eingestellt werden. Im folgenden Beispiel wird der Trace über den Befehl eines Teileprogramms gestartet. Die Variable ist über den Softkey "Quick list" auswählbar.

Wählen Sie die Option $AN_SLTRACE==1, um den Trace mit einer Variablen zu starten:

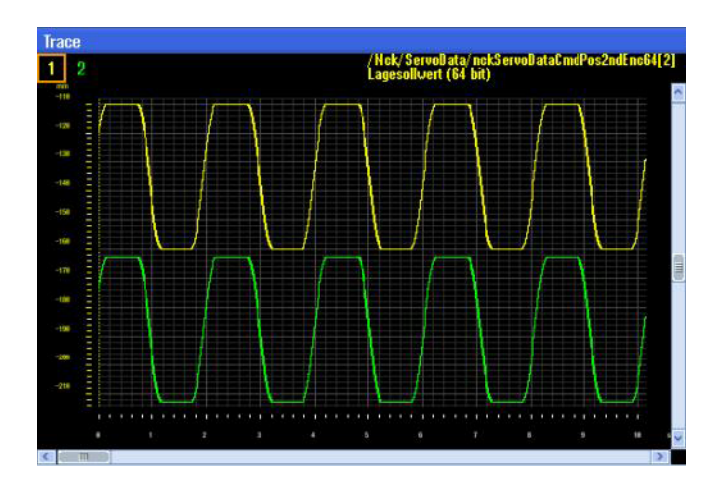


Es ist außerdem möglich, den Trace aus einem Wert der zuvor ausgewählten Signale auszulösen oder mit einer Systemvariablen, die über den Softkey "Variable einfügen“ verfügbar ist:



Ergebnis des Trace

Die Achse bewegt sich und die ausgewählten Signale werden für 5 Sekunden überwacht. Anschließend wird das Ergebnis angezeigt:

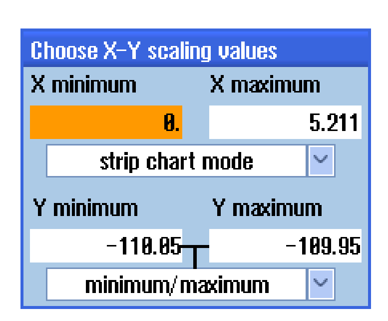


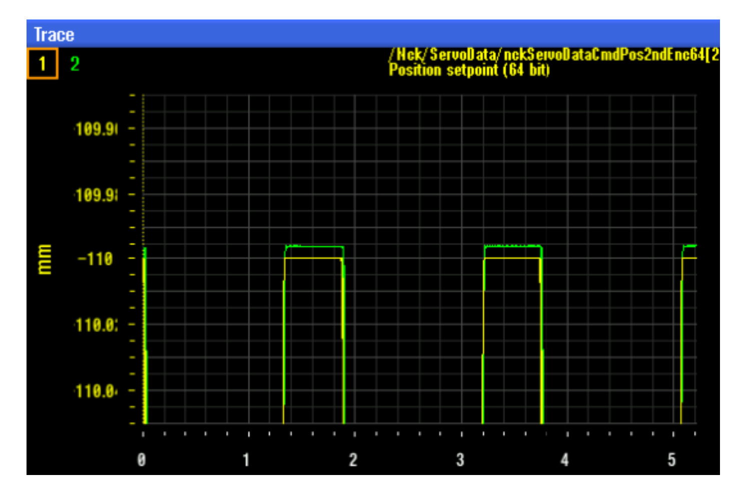
Trace-Maßstab einstellen

Es ist notwendig, den Teil des Trace einzusehen, in dem die Achse die Sollposition erreicht. Um dies im Detail zu sehen, stellen Sie den Trace-Maßstab ein: Softkey "Trace anzeigen" → Softkey "Maßstab".

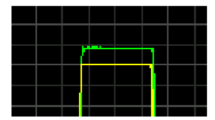
Das folgende Beispiel zeigt die Einstellung des Maßstabs, um die Achspositionierung bei -110 mm zu sehen. Die eingegebenen Werte führen zu einem 100-Mikrometer-Fenster rund um die Zielposition von -110 mm. Je nach verwendeter Maschine/Achse kann ein abweichender Maßstab notwendig sein.

Die X Minimum-/X Maximum-Werte können bei Bedarf ebenfalls angepasst werden:





Ansicht im Detail:



Sollposition (gelb) und Istposition (grün) stimmen nicht überein, da der Lageregler nicht aktiv ist.

7.3.4 Optimieren der Ruckeinstellung

Zielsetzung der Optimierung

Der Ruck muss so eingestellt werden, dass die Istpositon nicht überschritten wird. Dies passiert mit einem zu hoch eingestellten Ruckwert.

Ein zu niedrig eingestellter Ruckwert kann ebenfalls nachteilig für die Maschine sein, da er sich auf die Zeit auswirkt, die für die Positionierung einer Achse benötigt wird. Ohne den Lageregler wird die Achse ohne Überschwingen positioniert.

Der verbleibende Positionierfehler resultiert daraus, dass der Lageregler deaktiviert ist und die Achse sich lediglich mit Vorschubsignalen bewegt. Ein Überschwingen beim Erreichen der Position resultiert aus der Mechanik und kann durch Verwenden einer Ruckbegrenzung beseitigt werden.

Zuerst werden folgende Maschinendaten überprüft und eingestellt:

- MD32200 $MA_POSCTRL_GAIN = 0
- MD32610 $MA_VELO_FFW_WEIGHT = 1
- MD32620 $MA_FFW_MODE = 3
- MD32810 $MA_EQUIV_SPEEDCTRL_TIME = wie optimiert

Die folgenden Beispiele zeigen die Auswirkungen unterschiedlicher Ruckwerte auf die Positionierung einer Achse.

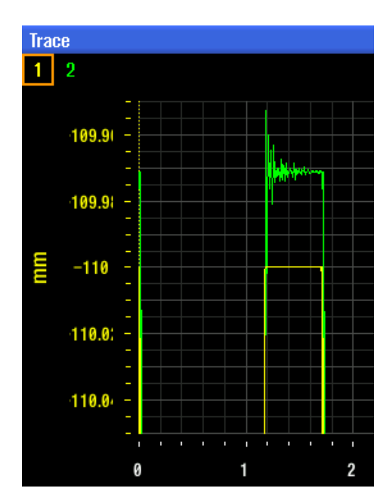
Beispiele

Für die Einstellung des Rucks wird das Achs-Maschinendatum MD32431 verwendet.

Beispiel 1: MD32431 $MA_MAX_AX_JERK = 600 ⇒ Ruck zu hoch.

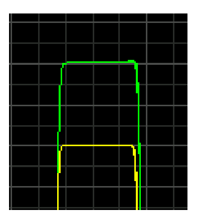
1. Starten Sie den Trace neu und führen Sie dann das Teileprogramm aus.
2. Wählen Sie den Softkey "Trace starten"; danach drücken Sie die Tasten <RESET> und <CYCLE START>.
3. Beseitigen Sie das Überschwingen durch Erhöhen des MAX_AX_JERK.

Dieses Diagramm zeigt den Ruckanstieg bis zu einem Überschwingen aufgrund der Mechanik.



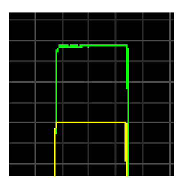
Beipiel 2: MD32431 $MA_MAX_AX_JERK = 2 ⇒ Ruck zu niedrig.

Bei einem zu niedrigen Ruck wird die Positionierkontur gerundet.



Beipiel 3: MD32431 $MA_MAX_AX_JERK = 65 ⇒ Ruck optimiert.

Ist der Ruck optimiert, kommt es zu geringem oder keinem Überschwingen und die Positionierkontur ist scharf.



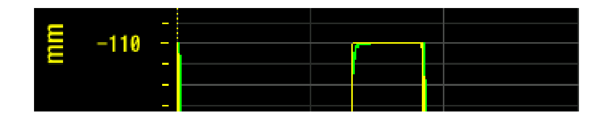
Beispiel: Positionierung mit Vorsteuerung und aktivem Lageregler

Wurde der Ruck optimiert, müssen Lageregler sowie Vorsteuerung aktiviert werden.

Stellen Sie die Maschinendaten nach Bedarf ein:

- MD20150[23] =2 Die Vorsteuerung kann dann als Voreinstellung freigegeben werden.
- MD32200 $MA_POSCTRL_GAIN = wie optimiert
- MD32610 $MA_VELO_FFW_WEIGHT = 1.0
- MD32620 $MA_FFW_MODE = 3
- MD32810 $MA_EQUIV_SPEEDCTRL_TIME = wie optimiert
- MD32431 $MA_MAX_AX_JERK = 65
- MD36012 $MA_STOP_LIMIT_FACTOR = Ursprungswert
- MD36400 $MA_CONTOUR_TOL = Ursprungswert

Eine gute Positionierung ohne Überschwingen erhält man, wenn Vorsteuerung und Ruck richtig optimiert sind:



7.4 Drehmomentauslastung

Maximale Achsbeschleunigung überprüfen

Um sicherzustellen, dass die Drehmomentgrenze nicht erreicht wird, überprüfen Sie nach der Optimierung des Kv-Faktors und der Vorschubvorsteuerung die Drehmomentauslastung unter den ungünstigsten Bedingungen:

- bei hoher Geschwindigkeit
- bei hoher Beladung

Die Funktion "Trace" wird dazu verwendet, um den Geschwindigkeitssollwert des aktiven Messsystems und die prozentuale Drehmomentuslastung über das Trace Signal "Auslastung (m_soll / m_soll_grenz)" zu messen. Wenn das Trace Signal "Auslastung (m_soll / m_soll_grenz)" = 100% ist, dann ist der Antrieb auf Strombegrenzung. Ändern Sie die Messparameter im Trace, um die Drehmomentauslastung anzuzeigen.

MD32300 $MA_MAX_AX_ACCEL definiert die maximale Beschleunigung der Achsen. Die Voreinstellung ist [1 m/s^2 (metrisch), 39,37 in/s^2 (inch) und 2,77 U/s^2 (rund)]. Jede Achse kann eine unterschiedliche Einstellung für die Beschleunigung haben.

Wenn der Maschinenhersteller es erlaubt, kann MD32300 so eingestellt werden, dass die Drehmomentauslastung während der Beschleunigungsphase im schlimmsten Fall zwischen 80% und 90% des Grenzwerts liegt, je nachdem wie viel Sicherheitsfaktor erwünscht ist.

Hinweis

Die Maschinenmechanik begrenzt die maximale Achsbeschleunigung. Der Maschinenhersteller muss den gewünschten Einstellungswert definieren! Wenn dieser Wert nicht definiert ist, verwenden Sie die Voreinstellung - außer der Maschinenhersteller erlaubt eine Erhöhung des Werts.

Beispiel: Testprogramm

In diesem Testprogramm sollen die Achsen innerhalb einer ausreichenden Distanz verfahren werden, so dass die programmierte Eilganggeschwindigkeit erreicht werden kann, z. B. 300 mm. Ändern Sie dazu das Testprogramm wie folgt:

```
SOFT

$AN_SLTRACE=0; Reset Start Servo-Trace-Trigger

LAB:

G0 X10

$AN_SLTRACE=1; Start Servo-Trace-Trigger

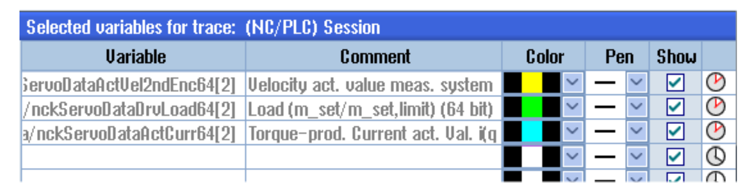
X310

GOTOB LAB

M30
```

Wählen Sie die folgenden Signale für die Überwachung aus:

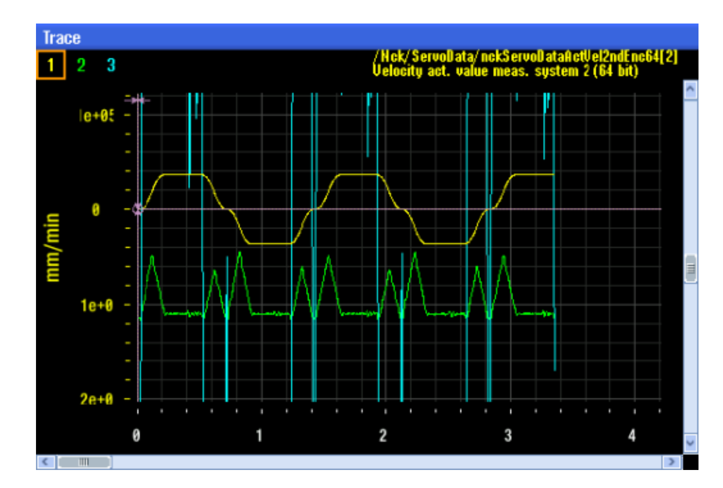
- Geschwindigkeitsistwert Messsystem 2
- Auslastung (m_soll/m_soll,grenz)
- Momentenbildender Stromistwert i(q)



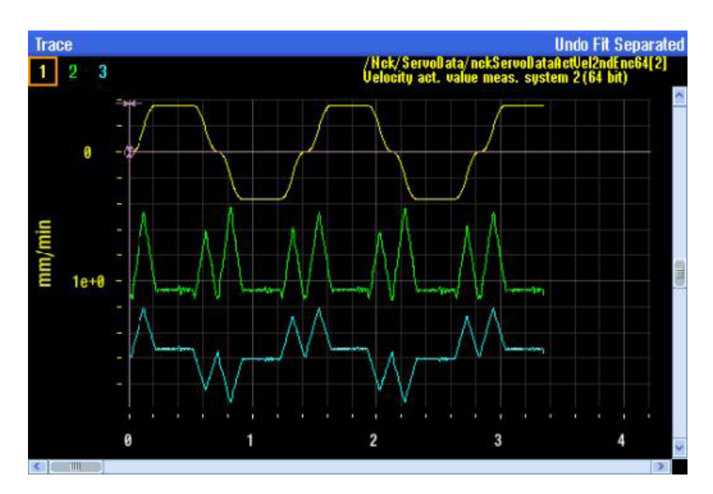
Selected variables for trace: (NC/PLC) Session

Variable	Comment	Color	Pen	Show
ServoDataActVel2ndEnc64[2]	Velocity act. value meas. system			
/nckServoDataDrvLoad64[2]	Load (m_set/m_set,limit) (64 bit)			
a/nckServoDataActCurr64[2]	Torque-prod. Current act. Val. i(q			

Softkey "Trace starten" → <RESET> → <CYCLE START>:



Die drei Traces werden gemeinsam dargestellt; drücken Sie den Softkey "Alle anpassen“:

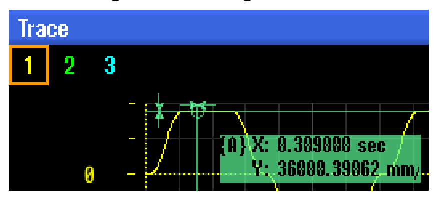


Um die Maximalwerte der drei Traces zu erkennen, drücken Sie den Softkey "Cursors" und positionieren Sie diese jeweils auf den Traces, um die Maximalwerte zu markieren: Softkey "Cursors" → Softkey "Cursor A".

Um die Maximalwerte der drei Traces zu überprüfen, drücken Sie den Softkey "Cursors" und positionieren Sie den Cursor auf den jeweiligen Trace, um die Maximalwerte folgender Größen abzulesen:

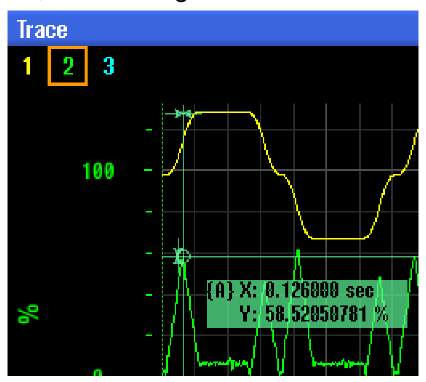
- Geschwindigkeit

  Der über dem Trace positionierte Cursor zeigt, dass die erreichte Maximalgeschwindigkeit 36 m/min beträgt:



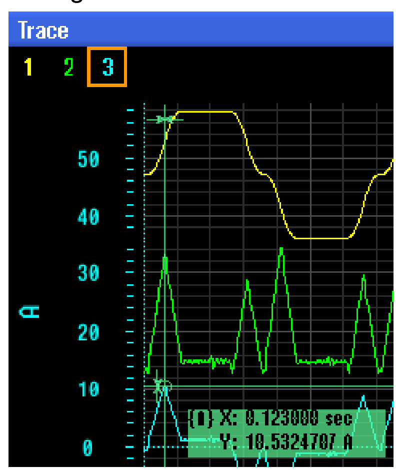
- Auslastung

  Der über dem Trace positionierte Cursor zeigt, dass die erreichte Maximalauslastung 58,52 % beträgt:



- Strom

  Der über dem Trace positionierte Cursor zeigt, dass der erreichte Maximalstrom 10,53 A beträgt:



Zusammenfassung

Die Maximalgeschwindigkeit von 36 m/min wird mit einer Maximalauslastung von 58,52% erreicht. Die Signalauswahl "Auslastung (m_soll/m_soll,grenz)" zeigt den Auslastungsprozentsatz der aktiven Stromgrenze. Die aktive Stromgrenze ist: p0640 = [Effektivwert A]

p0640 ist begrenzt auf p0338 Motorstromgrenze/ Mot I Max [Effektivwert A] oder r0209 Antriebsgrenzstrom/ PU I Max [Effektivwert A], je nachdem, welcher Wert niedriger ist.

In diesem Beispiel war p0640 = 18 A

Die Auslastung beträgt deshalb 18 A x 58,52/100 = 10,53 A

Die Prüfung wurde auf der Y-Achse einer Fräsmaschine mit einem Teilapparat auf dem Tisch durchgeführt. Die Auslastung von 58% erlaubt es ohne Probleme zusätzliches Gewicht hinzuzufügen. Die Auslastung sollte maximal 85% betragen. Um die Auslastung zu beeinflussen, kann die Beschleunigung verändert werden.

7.5 Kreisformtest

7.5.1 Kreisformtest: Funktion

Der Kreisformtest dient zum Einstellen und Beurteilen der Dynamik bei interpolierenden Achsen und zur Analyse der mittels der Reibkompensation (konventionelle Quadrantenfehlerkompensation) erzielten Konturgenauigkeit an den Quadrantenübergängen (kreisförmigen Konturen).

Der Kreisformtest wird dazu verwendet, um die Interpolation der zusammen arbeitenden Achsen zu überprüfen. Diese Funktion misst einen Kreis mit dem Verweis auf den Motor oder auf das direkte Messsystem. Eine Ausrichtung der Maschinen-Mechanik ist nicht im Ergebnis berücksichtigt. Dies gibt dem Inbetriebnehmer die Möglichkeit, Probleme der Regleroptimierung von mechanischen Problemen klar zu trennen.

Folgende Achs-Maschinendaten und Parameter werden mit diesem Verfahren überprüft:

- MD32200, MD32400, MD32402, MD32410, MD32490, MD32500, MD32510, MD32520, MD32540 MD32620, MD32640, MD32810, MD32900, MD32910, MD32930, MD32940
- p1421 bis p1426, p1400, p1433, p1434

Hinweis

MD32450 Umkehrlose muss über ein externes Gerät wie z. B. Kreisformtest oder Messuhr angepasst werden.

Die folgenden Positionsfehler-Kompensationen sollten abgeschaltet werden, wenn dieses Verfahren durchgeführt wird:

- MD32450 Umkehrlose-Kompensation
- MD32500 Reibkompensation aktiv
- MD32700 Geber-/Spindelfehler-Kompensation
- MD32710 Freigabe der Durchhangkompensation
- MD32750 Temperaturkompensationstyp

Beispiel

NC-Testprogramm für die Messung der X-Y-Achsen:

```
FFWON

SOFT

G90 G01 F3000 X400 Y200 Z500

LAB:

G91 G64 G02 X0 Z0 I10

GOTOB LAB

M30
```

Position, Vorschub und aktive Ebene müssen an die Maschine angepasst werden!

Spezifikation und Ergebnisse

Die besten Konturergebnisse werden erzielt, wenn die Kreisform-Testergebnisse in der richtigen tatsächlichen Größe, Form und minimaler p/p Abweichung zwischen einer kombinierten Interpolation der Achsen (X-Y, X-Z, Y-Z) liegen.

Ein NC-Programm in der Betriebsart MDA und die Kreisformtest-Funktion werden zur Messung und Bewertung dieser Ergebnisse verwendet werden. Der "schlimmste Fall" eines Kreisradius und der Bahngeschwindigkeit müssen zu einer realistischen Radialbeschleunigung führen, zu welcher die Maschine in der Lage ist.

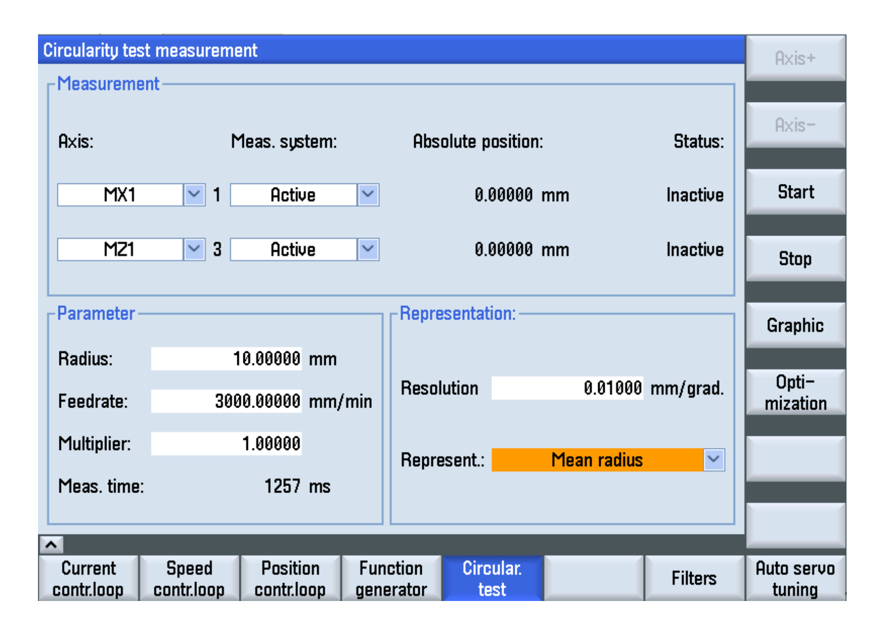
Maschinenhersteller haben in der Regel Spezifikationen für den Testkreis Radius und Vorschub.

Kreisformtests von Maschinenherstellern verwenden normalerweise einen Radius von 100 mm oder 150 mm bei Vorschubgeschwindigkeiten, die vom Maschinenhersteller bestimmt werden. Der Maschinenhersteller legt die Spezifikationen eines akzeptablen Ergebnisses fest.

High Speed-Bearbeitungen stellen in der Regel höhere Anforderungen beim Testen von Kreisen bei High-Speed-Fräsmaschinen und können im Bereich von 10 - 25 mm Kreisradius und Vorschüben von 5 - 10 m/min liegen. Für High-Speed-Fräsmaschinen, sind die Ergebnisse in der Regel akzeptabel, wenn die p/p Abweichung ≤ 0,010 mm und die tatsächliche Größe des Kreises gleich dem programmierten Radius, im schlimmsten Fall der Bahngeschwindigkeit ist.

7.5.2 Kreisformtest: Messung durchführen

Parameter einstellen



Um eine Messung durchzuführen, geben Sie folgende Parameter ein:

- **"Messung":** Auswahl der beiden Achsen, die vermessen werden sollen, und des Messsystems.
- **"Parameter":** Bei der Parametrierung der Eingabefelder "Radius" und "Vorschub" sind die entsprechenden Werte aus dem Teileprogramm, das die Kreisbewegung der Achsen steuert, unter Berücksichtigung des Vorschubkorrekturschalters einzutragen.
- **"Darstellung":** Parameter zur Anzeige der Grafik
  - "Auflösung" (Skalierung) der Diagrammachsen in [mm/Skt]
  - "Darstellung" über mittleren Radius oder programmierten Radius

Messung durchführen

Vorgehensweise:

1. Wählen Sie im Bedienbereich "Inbetriebnahme" den Softkey "Optimierung/Test" → Softkey "Kreisformtest".
2. Wählen Sie die Achsen, die gemessen werden sollen, mit der Taste <SELECT> oder über die Softkeys "Achse+"/"Achse -".
3. Stellen Sie die "Parameter" für die Messung ein: "Radius" und "Vorschub"

   Im Anzeigefeld "Messzeit" wird die aus den Werten "Radius" und "Vorschub" errechnete Messdauer für die Aufzeichnung der Lageistwerte bei der Kreisfahrt angezeigt:

   Ist die Messzeit ist nicht ausreichend, werden nur Teile vom Kreis dargestellt. Durch Reduzierung des Vorschubwertes kann die Messzeit erhöht werden. Dies trifft auch zu, wenn der Kreisformtest aus dem Stillstand gestartet wird.
4. Stellen Sie die Parameter zur Anzeige der Grafik ein:

   Überschreitet die daraus ermittelte Messzeit den darstellbaren Zeitbereich (maximale Messzeit = Lagereglertakt * 2048), so wird für die Aufzeichnung eine entsprechende gröbere Abtastung (n * Lagereglertakt) vorgenommen, damit ein vollständiger Kreis darstellbar ist.

Weitere Aktionen:

- Um die Messung zu starten, drücken Sie den Softkey "Start".
- Um die Messung zu stoppen, drücken Sie den Softkey "Stopp".
- Um weitere Anpassungen zur Optimierung durchzuführen, drücken Sie den Softkey "Optimierung".
- Auf einer neuen Softkey-Leiste können Sie **direkt** zu folgenden Bereichen navigieren:
  - "Service Achse" im Bedienbereich "Diagnose"
  - "Achs-Maschinendaten"
  - "Antriebs-Maschinendaten"
  - "Anwendersichten"

- Um die Parameter für eine Messung zu speichern, drücken Sie den Softkey "Parameter sichern".
- Um z. B. eine Messung mit denselben Parametern zu wiederholen, drücken Sie den Softkey "Parameter laden".

Grafik anzeigen

Um das Messergebnis als Grafik anzuzeigen, drücken Sie den Softkey "Grafik".

7.5.3 Kreisformtest: Beispiele

Die axiale Ruckbegrenzung MD32400 $MC_AX_JERK_ENABLE wird über eine Zeitkonstante eingestellt und ist immer aktiv.

Maschinendaten zum Lagesollwertfilter:

- MD32402 $MC_AX_JERK_MODE = Typ 2 wird empfohlen, Typ 1 ist aus Kompatibilitätsgründen voreingestellt. Die Parametrierung einer reinen Bandsperre wird ausdrücklich nicht empfohlen.
- MD32402 $MA_AX_JERK_MODE (Filtertyp) und MD32410 $MA_AX_JERK_TIME > 0 ist nur wirksam, wenn MD32400 $MA_AX_JERK_ENABLE = 1 gesetzt ist.

Beispiel 1 zur Optimierung

Maschinendaten nach der Optimierung der Achsen:

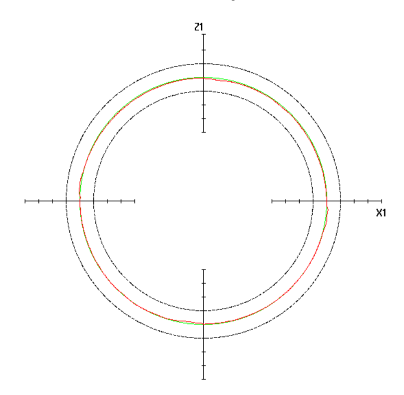
Parameter/Maschinendaten		Achse X	Achse Z
MD32200	$MC_POSCTRL_GAIN	8.500	8.500
p1460	SPEEDCTRL_GAIN1	3.01	3.89
p1462	SPEEDCTRL_INTEGRATOR_TIME_1	6.18	6.18
p1463	SPEEDCTRL_REF_MODEL_FREQ	106.3	106.3
p1440	NUM_SPEED_FILTERS	0	0
MD32610	$MC_VELO_FFW_WEIGHT	1.0	1.0
MD32620	$MC_FFW_MODE	4	4
MD32810	$MC_EQUIV_SPEEDCTRL_TIME	0.0022	0.0022
MD32400	$MC_AX_JERK_ENABLE	0	0

In der Regel ist mit optimierter Vorschubvorsteuerung der tatsächliche Wert des Radius ist zu groß. Dies kann mit der Zeitkonstante MD32410 $MC_AX_JERK_TIME korrigiert werden. Falls erforderlich, benutzen Sie eine Zeitkonstante in allen Achsen.

Dieser Kreis zeigt die Ergebnisse an, nachdem die Vorschubvorsteuerung optimiert wurde. Der mittlere Radius ist jedoch um 0,0019 mm zu groß:

X1: aktives Messsystem

Z1: aktives Messsystem



Parameter

Radius: 10.00000 mm

Vorschub: 3000.00000 mm/min

Messzeit: 1257 ms

X1: aktives Messsystem

Z1: aktives Messsystem

Darstellung

Auflösung: 0.01000 mm

Darstellung. mittlerer Radius

Radius: 10.00190 mm

Delta R: 4.02698 µm

Parameter/Maschinendaten		Achse X	Achse Z
MD32200	$MC_POSCTRL_GAIN	8.500	8.500
p1460	SPEEDCTRL_GAIN1	3.01	3.89
p1462	SPEEDCTRL_INTEGRATOR_TIME_1	6.18	6.18
p1463	SPEEDCTRL_REF_MODEL_FREQ	106.3	106.3
p1440	NUM_SPEED_FILTERS	0	0
MD32610	$MC_VELO_FFW_WEIGHT	1.0	1.0
MD32620	$MC_FFW_MODE	3	3
MD32810	$MC_EQUIV_SPEEDCTRL_TIME	0.0022	0.0022
MD32400	$MC_AX_JERK_ENABLE	1	1
MD32402	$MC_AX_JERK_MODE	2	2
MD32410	$MC_AX_JERK_TIME	**0.012**	**0.012**

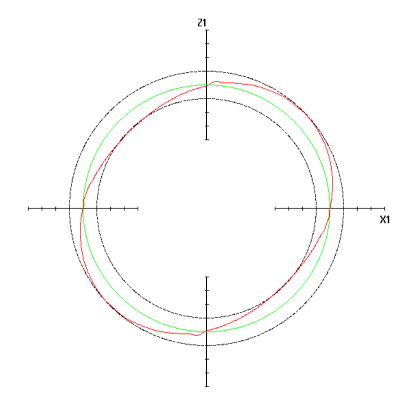
Beispiel 2 zur Optimierung

Dieser Kreis zeigt den Effekt einer geringfügig unterschiedlichen Zeitkonstante für den axialen Ruckfilter. Um diese Art von Fehler zu korrigieren, wird die Zeitkonstante angepasst:

Parameter/Maschinendaten		Achse X	Achse Z
MD32400	$MC_AX_JERK_ENABLE	1	1
MD32402	$MC_AX_JERK_MODE	2	2
MD32410	$MC_AX_JERK_TIME	**0.012**	**0.0125**

X1: aktives Messsystem

Z1: aktives Messsystem



Parameter

Radius:	10.00000 mm
Vorschub:	3000.00000 mm/min
Messzeit:	1257 ms
X1:	aktives Messsystem
Z1:	aktives Messsystem

Darstellung

Auflösung:	0.01000 mm
Darstellung:	mittlerer Radius
Radius:	10.00029 mm
Delta R:	25.47002 µm

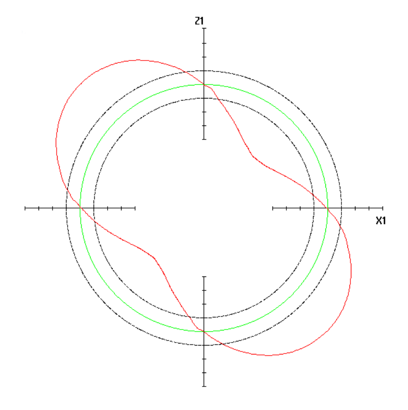
Beispiel 3 zur Optimierung

Dieser Kreis zeigt den Effekt einer deutlich unterschiedlichen Zeitkonstante für den axialen Ruckfilter. Um diese Art von Fehler zu korrigieren, wird die Zeitkonstante angepasst:

Parameter/Maschinendaten		Achse X	Achse Z
MD32400	$MC_AX_JERK_ENABLE	1	1
MD32402	$MC_AX_JERK_MODE	2	2
MD32410	$MC_AX_JERK_TIME	**0.015**	**0.012**

X1: aktives Messsystem

Z1: aktives Messsystem



Parameter

Radius:	10.00000 mm
Vorschub:	3000.00000 mm/min
Messzeit:	1257 ms
X1:	aktives Messsystem
Z1:	aktives Messsystem

Darstellung

Auflösung:	0.01000 mm
Darstellung:	mittlerer Radius
Radius:	9.98971 mm
Delta R:	75.67665 µm

7.5.4 Kreisformtest: Daten sichern

Folgende Daten können Sie beim Kreisformtest sichern:

- **Parameter sichern:** Die eingegebenen Parameter können als Datei gespeichert werden.

  Es ist folgender Pfad
  voreingestellt: user/sinumerik/hmi/log/optimization/cicular/<name>.sup

- **Grafik sichern:** Wurde die Grafik korrekt gespeichert, so erscheint eine Meldung "Daten wurden gespeichert", ansonsten wird eine Fehlermeldung "Fehler beim Speichern der Datei" ausgegeben.

  Es ist folgender Pfad
  voreingestellt: user/sinumerik/hmi/log/optimization/cicular/<name>.sud

- **Grafik drucken:** Die Grafik wird als Pixelgrafik im Format PNG gespeichert. Geben Sie einen Namen <name> ein. Die Bezeichnung ist frei wählbar. Es ist folgender Pfad voreingestellt: user/sinumerik/hmi/log/optimization/cicular/<name>.png

Parameter sichern

Das Dateiformat ist wie folgt strukturiert (mit Kommentaren in Klammern[ ])

```
H: CstPar [Kennung für Paramter Kreisformtest]

V: 5.0 [Versionsnummer des Dateiformats]

@Messung

P 1: 1 [Achsnummer erste Achse]

P 2: 2 [Achsnummer zweite Achse]

P 3: 0 [Messsystem Achse 1 - 0:aktives ; 1:erstes ; 2: zweites]

P 4: 0 [Messsystem Achse 2 - 0:aktives ; 1:erstes ; 2: zweites]

@Parameter

P 10: 30  [Radius]

P 11: 3000 [Vorschub]

P 12: 3770 [Messzeit]

P 14: 1 [Multiplikator]

@Darstellung

P 20: 10  [Auflösung]

P 21: 8 [mittlerer / prog. Radius - 8 = mittlerer R ; 9 = prog. Radius]
```

Grafik sichern

Das Dateiformat ist wie folgt strukturiert (mit Kommentaren in Klammern[ ]):

```
H: CstPic [Kennung für Grafik Kreisformtest]

V: 5.0 [Versionsnummer des Dateiformats]

@Parameter

P 1: 30 [Radius]

P 2: 3000 [Vorschub]

P 3: 3770 [Messzeit]

P 4: 0 [Messsystem Achse 1 - 0:aktives ; 1:erstes ; 2: zweites]

P 5: 0 [Messsystem Achse 2 - 0:aktives ; 1:erstes ; 2: zweites]

@Darstellung

P 10: 10 [Auflösung]

P 11: 9 [mittlerer / prog. Radius - 8 = mittlerer R ; 9 = prog. Radius]

P 12: X1 [Achsname 1]

P 13: Z1 [Achsname 2]

@Zwischenwerte

P 20: 15.6632 [max. Radius Messwerte]
```

```
P 21: 10.9326 [min. Radius Messwerte]

P 22: 13.6694 [mittl. Radius Messwerte]

P 23: 1886 [Anzahl Messwerte]

@Zusatzwerte

P 30: 1000 [Genauigkeit (1/P30))

@Physikalische Einheiten

P 40: 5370 [Textnummer Einheit Radius]

P 41: 5381 [Textnummer Einheit Vorschub]

P 42: 6165 [Textnummer Einheit Auflösung]

P 43: 5346 [Textnummer Einheit DeltaRadius]

P 44: 0 [Neu: Operate: Basislengthunit]

@Abszisse

Ai:   [Abzissenwerte i : 0..P23]

@Ordinate

Oi:   [Ordinatenwerte i : 0..P23]

@Radius

Ri:   [Radiuswerte i : 0..P23]
```

7.6 Spindel optimieren

7.6.1 Maschinendaten für die Spindel einstellen

Einstellung der Maschinendaten prüfen

Gleichen Sie die voreingestellten Maschinendaten sowie alle relevanten Parameter für Spindel-/Achsinterpolation mit den MD in der folgenden Tabelle ab. Das Beispiel gilt für einen Spindelmotor ohne mechanische Getriebestufen mit 8000 U/min.

Die Maschinendaten können mit den in der Tabelle vorgeschlagenen Werten voreingestellt werden. Die Werte variieren je nach Anwendungsfall.

Nummer	Bezeichnug	Wert	Bedeutung
MD30300	$MA_IS_ROT_AX	1	Rundachse / Spindel
MD30310	$MA_ROT_IS_MODULO	1	Modulowandlung für Rundachse / Spindel
MD30320	$MA_DISPLAY_IS_MODULO	1	Modulo 360 Grad Anzeige bei Rundachse oder Spindel
MD32000	$MA_MAX_AX_VELO	8000	maximale Achsgeschwindigkeit
MD32010	$MA_JOG_VELO_RAPID	60	Konventioneller Eilgang
MD32020	$MA_JOG_VELO	10	Konventionelle Achsgeschwindigkeit
MD32040	$MA_JOG_REV_VELO_RAPID	60	Umdrehungsvorschub bei JOG mit Eilgangsüberlagerung
MD32050	$MA_JOG_REV_VELO	10	Umdrehungsvorschub bei JOG
MD32200[0]	$MA_POSCTRL_GAIN	x	K_V-Faktor: wie optimiert
MD32200[1]	$MA_POSCTRL_GAIN	x	
MD32620	$MA_FFW_MODE	3	Vorsteuerungsart
MD32640	$MA_STIFFNESS_CONTROL_ENABLE	1	Dynamische Steifigkeits-Regelung
MD32810[0]	$MA_EQUIV_SPEEDCTRL_TIME	x	Ersatzzeitkonstante Drehzahlregelkreis für Vorsteuerung: wie optimiert
MD32810[1]	$MA_EQUIV_SPEEDCTRL_TIME	x	
MD33000	$MA_FIPO_TYPE	3	Feininterpolatortyp
MD34000	$MA_REFP_CAM_IS_ACTIVE	0	Achse mit Referenzpunktnocken
MD34020	$MA_REFP_VELO_SEARCH_CAM	30	Referenzpunktanfahrgeschwindigkeit
MD34020	$MA_REFP_VELO_SEARCH_CAM	30	
MD34040[0]	$MA_REFP_VELO_SEARCH_MARKER	30	Abschaltgeschwindigkeit
MD34040[1]	$MA_REFP_VELO_SEARCH_MARKER	30	
MD34060[0]	$MA_REFP_MAX_MARKER_DIST	370	maximale Wegstrecke zur Referenzmarke
MD34060[1]	$MA_REFP_MAX_MARKER_DIST	370	
MD35000	$MA_SPIND_ASSIGN_TO_MACHAX	1	Zuordnung Spindel zu Maschinenachse
MD35100	$MA_SPIND_VELO_LIMIT	8000	Maximale Spindeldrehzahl
MD35110[0]	$MA_GEAR_STEP_MAX_VELO	8000	Maximaldrehzahl für Getriebestufenwechsel
MD35110[1]	$MA_GEAR_STEP_MAX_VELO	8000	

Nummer	Bezeichnug	Wert	Bedeutung
MD35130[0]	$MA_GEAR_STEP_MAX_VELO_LIMIT	8000	Maximaldrehzahl der Getriebestufe
MD35130[1]	$MA_GEAR_STEP_MAX_VELO_LIMIT	8000	
MD35200[0]	$MA_GEAR_STEP_SPEEDCTRL_ACCEL	x	Beschleunigung im Drehzahlregelbetrieb: wie optimiert
MD35200[1]	$MA_GEAR_STEP_SPEEDCTRL_ACCEL	x	
MD35210[0]	$MA_GEAR_STEP_POSCTRL_ACCEL	x	Beschleunigung im Lageregelbetrieb: wie optimiert
MD35210[1]	$MA_GEAR_STEP_POSCTRL_ACCEL	x	
MD35500	$MA_SPIND_ON_SPEED_AT_IPO_START	2	Vorschubfreigabe bei Spindel im Sollbereich
MD35550[0]	$MA_DRILL_VELO_LIMIT	4000	Maximaldrehzahlen für das Gewindebohren
MD35550[1]	$MA_DRILL_VELO_LIMIT	4000	
MD36000	$MA_STOP_LIMIT_COARSE	0,4	Genauhalt grob
MD36010	$MA_STOP_LIMIT_FINE	0,1	Genauhalt fein
MD36030	$MA_STANDSTILL_POS_TOL	5	Stillstandstoleranz
MD36040	$MA_STANDSTILL_DELAY_TIME	1	Verzögerungszeit Stillstandsüberwachung
MD36050	$MA_STOP_ON_CLAMPING	1	Klemmungstoleranz
MD36060	$MA_STANDSTILL_VELO_TOL	2	Schwellgeschwindigkeit/Drehzahl "Achse/Spindel steht"
MD36200	$MA_AX_VELO_LIMIT	8800	Schwellwert für Geschwindigkeitsüberwachung
MD36200	$MA_AX_VELO_LIMIT	8800	
MD36300[0]	$MA_ENC_FREQ_LIMIT	1000000	Geber Grenzfrequenz
MD36300[1]	$MA_ENC_FREQ_LIMIT	1000000	
MD36400	$MA_CONTOUR_TOL	30	Toleranzband Konturüberwachung
p1433	SPEED_CONTROLLER_REFERENCE_ MODEL_NATURAL_FREQUENCY	x	Drehzahlregler Referenzmodell Eigenfrequenz / n_reg RefMod fn: wie optimiert

7.6.2 Spindel: So prüfen Sie den Drehzahlregler

Beschleunigung des Drehzahlreglers prüfen

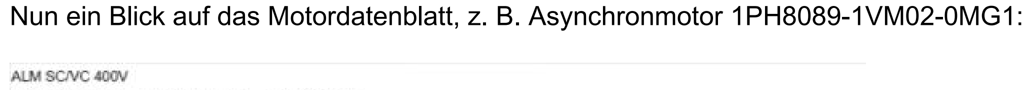
Es wird eine Überprüfung der Drehzahlreglerbeschleunigung vorgenommen. Zunächst muss die Beschleunigungszeit ohne die Beeinflussung durch die NC festgestellt werden.

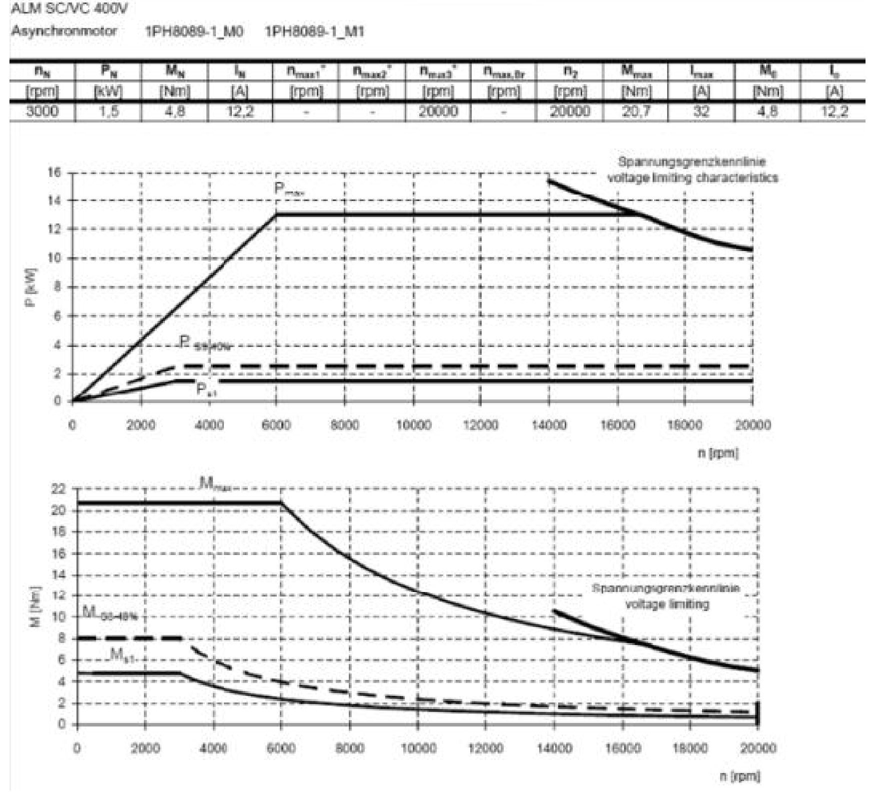
Setzen Sie folgendes Maschinendatum:

- MD35200[0] $MA_GEAR_STEP_SPEEDCTRL_ACCEL = 9999
- MD35200[1] $MA_GEAR_STEP_SPEEDCTRL_ACCEL = 9999

  (t=V/a t=9000/60s/9999 t=15ms daher keinen Einfluss).

Nun ein Blick auf das Motordatenblatt, z. B. Asynchronmotor 1PH8089-1VM02-0MG1:





ALM SC/VC 400V

Asynchronmotor 1PH8089-1_M0 1PH8089-1_M1

n_N	P_N	M_N	I_N	n_{max1}*	n_{max2}*	n_{max3}*	$n_{max,Br}$	n_2	M_{max}	I_{max}	M_0	I_0
[rpm]	[kW]	[Nm]	[A]	[rpm]	[rpm]	[rpm]	[rpm]	[rpm]	[Nm]	[A]	[Nm]	[A]
3000	1,5	4,8	12,2	-	-	20000	-	20000	20,7	32	4,8	12,2

Bild 7-3 Asynchronmotor 1PH8

Beschleunigen und Abbremsen mit den ursprünglichen Daten des Motors entspricht der S1-Kurve.

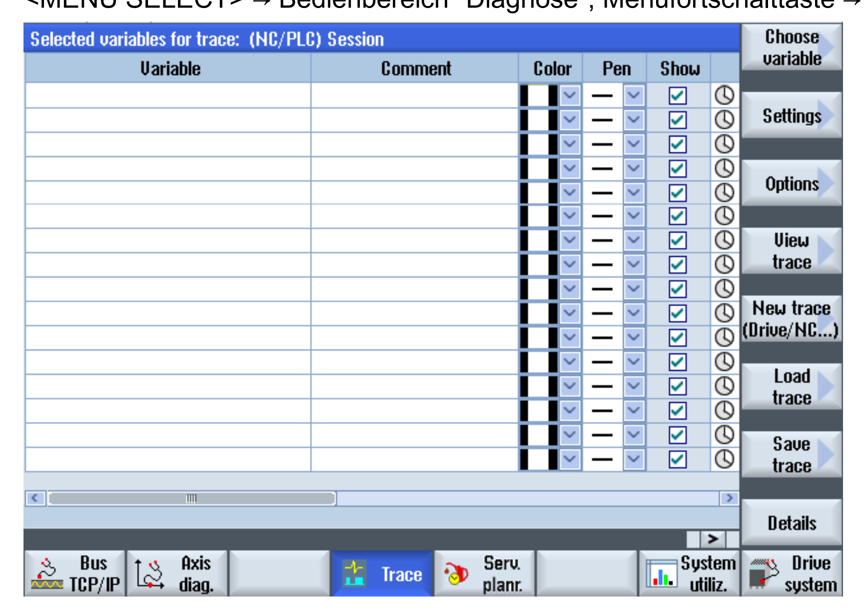
Nummer	Bezeichnung	Wert
p0640	Stromgrenze (p0305 Motornennstrom * 1,5)	17,7 A
p1520	CO: Drehmomentgrenze oben/motorisch / M_max oben/mot	4,8 Nm
p1521	CO: Drehmomentgrenze unten/generatorisch / M_max unten/gen	-4,8 Nm
p1530	Leistungsgrenze motorisch / P_max mot	1,5 kW
p1531	Leistungsgrenze generatorisch / P_max gen	-1,5 kW

Trace aufzeichnen

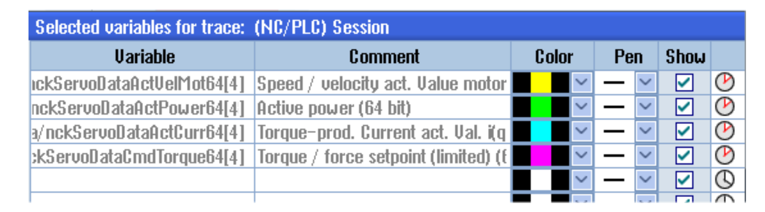
Vorgehensweise:

1. Verwenden Sie die folgende Tastenkombination, um die Funktion Trace aufzurufen:

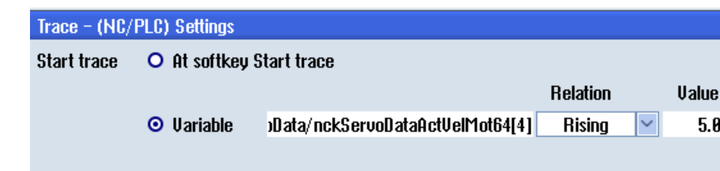
   <MENU SELECT> → Bedienbereich "Diagnose", Menüfortschalttaste → Softkey "Trace":



2. Um die benötigten Variablen auszuwählen: Softkey "variable wählen"
   - Drehzahl-/Geschwindigkeitsistwert Motor
   - Wirkleistung
   - Momentenbildender Stromistwert i(q)
   - Momenten-/Kraft-Sollwert (begrenzt)
3. Die benötigten Variablen werden ausgewählt: Softkey "Variable wählen".
4. Verwenden Sie den Softkey "Filter/Suchen", um die Liste der Variablen einzugrenzen.
5. Stellen Sie sicher, dass die Spindel ausgewählt ist, wenn Sie die Variablen hinzufügen:



6. Drücken Sie den Softkey "Einstellungen".



7. Bestätigen Sie mit "OK".
8. Geben Sie in MDA folgendes Programm ein:

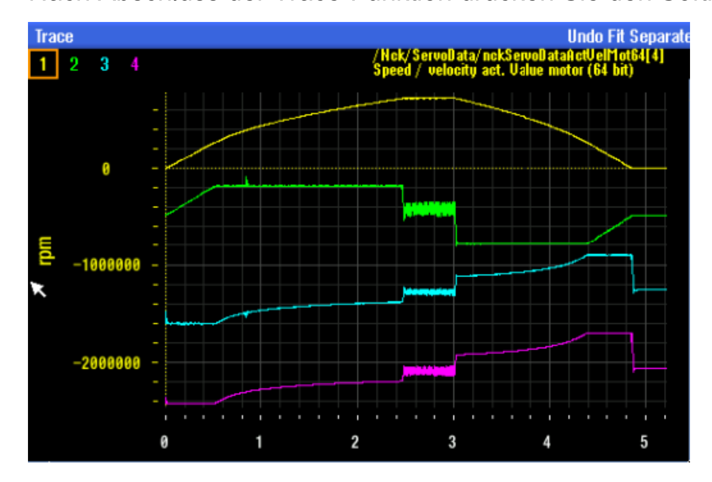
```
S8000 M03
G04 F5
M30
```

9. Initialisieren Sie den Trace und führen Sie das Programm aus:

   Softkey "Trace" → Softkey "Trace anzeigen" → Softkey "Trace starten" → Taste <RESET> → Taste <CYCLE START>

Auswertung

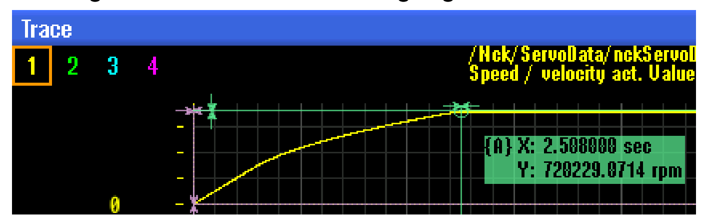
- Nach Abschluss der Trace-Funktion drücken Sie den Softkey "Alle anpassen“.



Zum Einstellen der Beschleunigungszeit verwenden Sie die Cursor-Tasten.

Trace-Auswahl: Istdrehzahl (Trace 1)

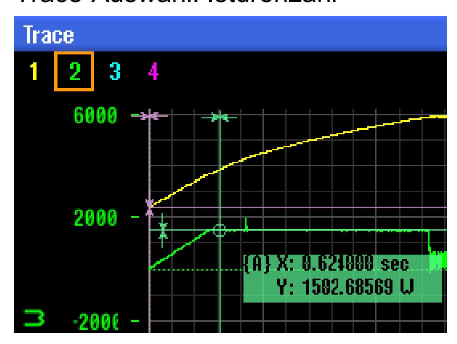
Benötigte Zeit für die Beschleunigung bis zur Solldrehzahl = 2,5 Sekunden



Der Cursor wird am Trace-Anfang positioniert und an dem Punkt, an dem die Solldrehzahl erreicht wird.

- Die Leistung, die für die Beschleunigung des Motors verwendet wird, ist auf Trace 2 einzusehen:

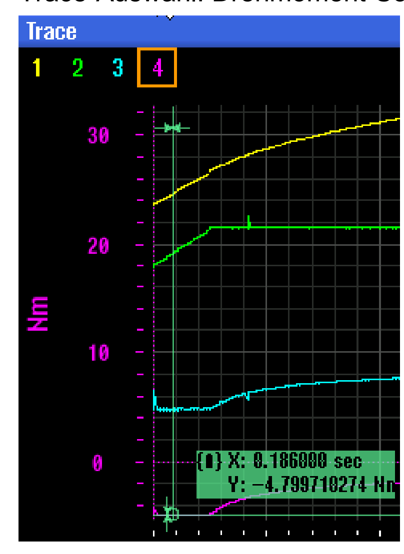
Trace-Auswahl: Istdrehzahl



Maximal verfügbare Leistung = 1,5 kW = S1-Kurve

- Der Drehmoment-Sollwert ist auf Trace 4 einsehbar:

Trace-Auswahl: Drehmoment-Sollwert



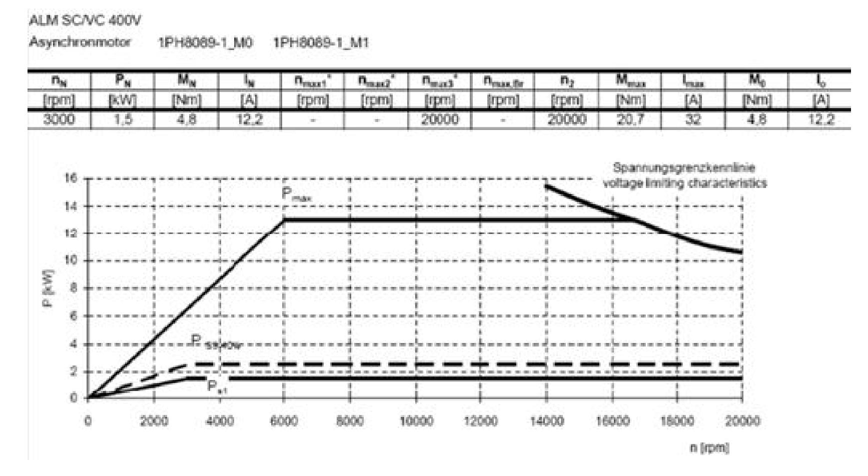
Drehmoment-Sollwert = 4,8 Nm = S1-Kurve

Beschleunigung prüfen

Die Beschleunigung/Abbremsung kann nun anhand des Motordatenblatts bis hin zu den Maximalwerten verändert werden; dies hängt jedoch von der jeweiligen Anwendung ab. Geben Sie die für die Anwendung passenden Werte ein. In unserem Beispiel war eine kurze Beschleunigungs-/Bremszeit gefordert.

Nummer	Bezeichnung	Wert
p0640	Stromgrenze (p0305 Motornennstrom * 1,5)	32 A
p1520	CO: Drehmomentgrenze oben/motorisch / M_max oben/mot	20,7 Nm
p1521	CO: Drehmomentgrenze unten/generatorisch / M_max unten/gen	-20,7 Nm
p1530	Leistungsgrenze motorisch / P_max mot	13 kW
p1531	Leistungsgrenze generatorisch / P_max gen	-13 kW

Die Maximalwerte des Motordatenblatts werden in die entsprechenden Antriebsdaten eingegeben: z. B. Asynchronmotor 1PH8089-1_M0, 1PH8089-1_M1


ALM SC/VC 400V
Asynchronmotor 1PH8089-1_M0 1PH8089-1_M1

n_N	P_N	M_N	I_N	n_{max1}	n_{max2}	n_{max3}	$n_{max,Br}$	n_2	M_{max}	I_{max}	M_0	I_0
[rpm]	[kW]	[Nm]	[A]	[rpm]	[rpm]	[rpm]	[rpm]	[rpm]	[Nm]	[A]	[Nm]	[A]
3000	1,5	4,8	12,2	-	-	20000	-	20000	20,7	32	4,8	12,2

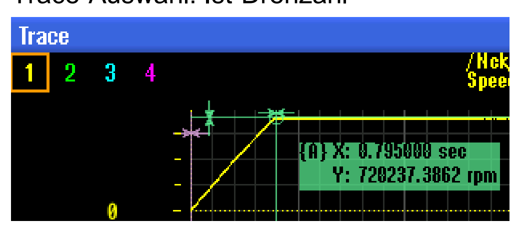
Bild 7-4 Asynchronmotor 1PH8

Neuen Trace aufzeichnen

Vorgehensweise:

1. Führen Sie die Spindel- und Trace-Funktion aus. Jetzt kann die benötigte Zeit für die Beschleunigung bis zur Solldrehzahl überprüft werden.

   Trace-Auswahl: Ist-Drehzahl

2. Stellen Sie dasMD35200so ein, dass es den Anforderungen des Maschinenherstellers für die Spindelbeschleunigung entspricht:

   MD35200[0] $MA_GEAR_STEP_SPEEDCTRL_ACCEL

   MD35200[1] $MA_GEAR_STEP_SPEEDCTRL_ACCEL

7.6.3 Spindel: So prüfen Sie den Lageregler

Test des Lagereglers

Es kann nun auch ein Test durchgeführt werden, um Beschleunigung und Ruck der Lageregelung einzustellen. Dieser Test erfolgt durch Gewindebohren ohne Ausgleichsfutter.

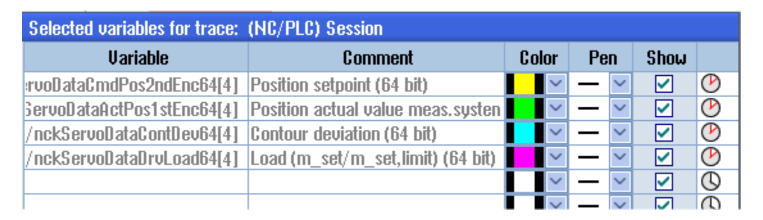
Vorgehensweise:

1. Überprüfen Sie MD35550 [0] & [1] DRILL_VELO_LIMIT; hierbei handelt es sich um die maximale Gewindebohrgeschwindigkeit.
2. Überprüfen Sie SD55484 $SCS_DRILL_TAPPING_SET_MC[0] = 1; es hält die Spindel während des Gewindebohrens im Positionierbetrieb.

   Es kann das folgende Programm verwendet werden, das 10 Löcher auf derselben Position bohrt:

```
SUPA D0 G0 G90 Z-200
FFWON
M19
SOFT
LAB:
G331 Z-214 S4000 K1
G332 Z-200 S4000 K1
REPEAT LAB P=9
M30
```

3. Wählen Sie folgende Variablen aus:
   - Lagesollwert (64 bit)
   - Lageistwert Messsystem (64 bit)
   - Konturabweichung (64 bit)
   - Auslastung (m_soll/m_soll,grenz) (64 bit)

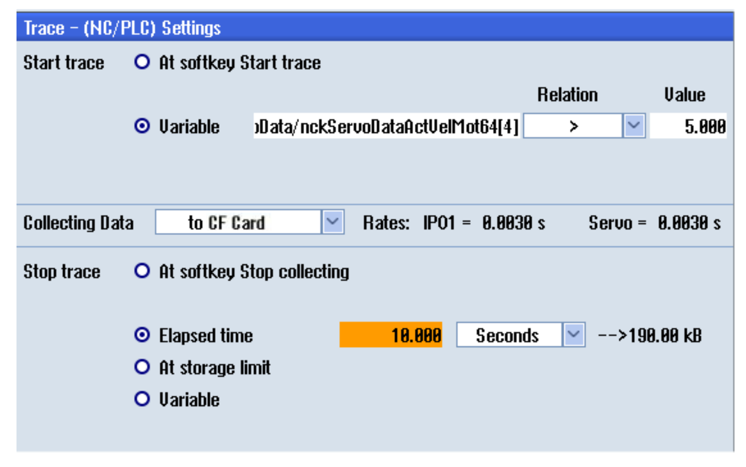


Hinweis

Stellen Sie sicher, dass bei der Überwachung von Messsystemen das richtige Signal ausgewählt ist, z. B. Messsystem 1 oder 2.

---

4. Wählen Sie den Trace-Trigger usw. über den Softkey "Einstellungen" aus. Das Beispiel zeigt einen Trace, der ausgelöst wird sobald die Istposition 5 Grad überschreitet. Der Trace wird für 10 Sekunden aufgezeichnet.



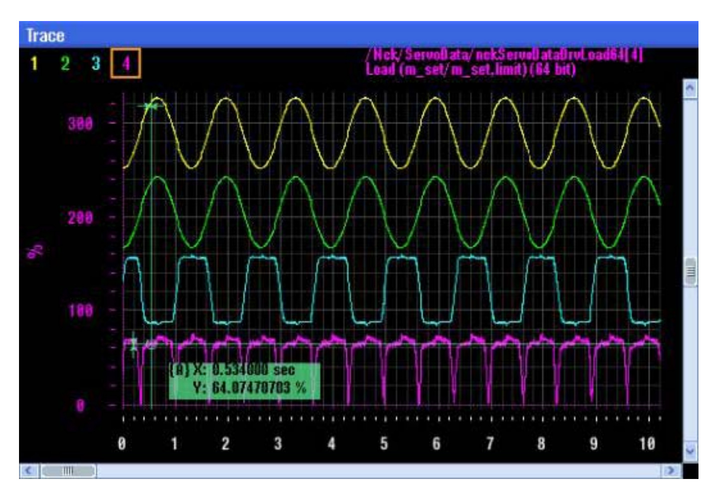
5. Starten Sie den Trace und drücken Sie "CYCLE START" auf der Maschinensteuertafel.

Beispiele

Im folgenden Beispiel erreicht die Auslastung 64%, weshalb die Lageregelungs-beschleunigung erhöht werden kann. Während des Gewindebohrens kann die Spindel jedoch oberhalb der S1-Kurve arbeiten, so dass sichergestellt werden muss, dass die Auslastung nicht 100% erreicht, da sonst Alarme zur Konturüberwachung ausgegeben werden.

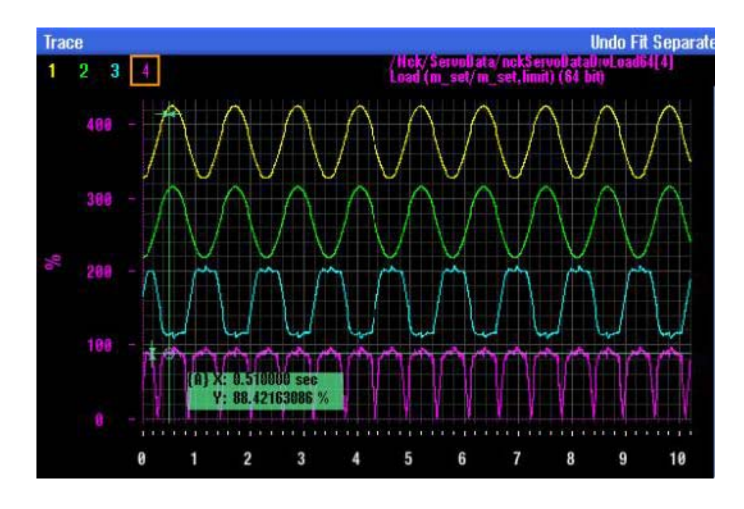
1. Optimieren Sie MD35210 $MA_GEAR_STEP_POSCTRL_ACCEL und MD32431 $MA_MAX_AX_JERK, um optimale Ergebnisse zu erhalten.
2. Überprüfen Sie außerdem den anpassungsfähigen Regelbereich p1464, p1465 der Spindel.

Spindel: Auslastung 64%



Das folgende Beispiel zeigt die Auswirkung einer Erhöhung der Lageregelungs-beschleunigung von 120 auf 160; die Auslastung steigt dann von 64% auf 88%. Handelt es sich bei der Anwendung um eine reine Gewindebohrmaschine, kann ein Test durchgeführt werden, bei dem 100 Löcher mit Gewinde gebohrt werden. Anschließend wird die Aufzeichnung des Trace wiederholt, um sicherzustellen, dass die Auslastung nie 100% erreicht.

Spindel: Auslastung 88%



8 Service Planner

Übersicht

Der Wartungsplaner ermöglicht es, mittels Wartungsplanerdialog an der Bedienoberfläche oder Programming Tool Zeitintervalle und Alarmsequenzen für zu bearbeitende Aufgaben (meist Aufgaben der Maschinenwartung) zu editieren, zu starten und zu de- oder reaktivieren.

Die numerischen Daten der Aufgaben werden in Datenbausteinen organisiert und in der Anwenderschnittstelle für HMI, PLC-Anwenderprogramm und Programming Tool zur Verfügung gestellt. Die Bezeichnungen der jeweiligen Aufgabe, werden von der Bediensoftware verwaltet, editiert und zusammen mit den numerischen Daten angezeigt.

Die PLC-Firmware greift ihrerseits auf die Datenbausteine der Anwenderschnittstelle zu, verarbeitet die Daten und stellt die Ergebnisse in Form von Restzeiten sowie Warnungen und Alarmen wiederum in Datenbausteinen zur Verfügung. Der Wartungsplaner wird in der PLC-Firmware minütlich bearbeitet. Beim Ausschalten der Steuerung werden die Istdaten der Wartungsaufgaben eingefroren. Beim Einschalten wird dann auf diese remanent gespeicherten Werte aufgesetzt.

Das PLC-Anwenderprogramm wertet die Istdaten aus und generiert Warn- oder Alarmmeldungen in numerischer Form wahlweise mit oder ohne Power-OFF-Status. Der Alarmhandler setzt diese Meldungen mit der entsprechenden PLC Alarmtextdatei oem_alarm_plc_<lng>.ts in eine Meldung für den Bediener um, die an der Bedienoberfläche angezeigt wird (<lng> aktuell eingestellte Sprache) und die bei Bedarf protokolliert werden kann.

Konfiguration im System

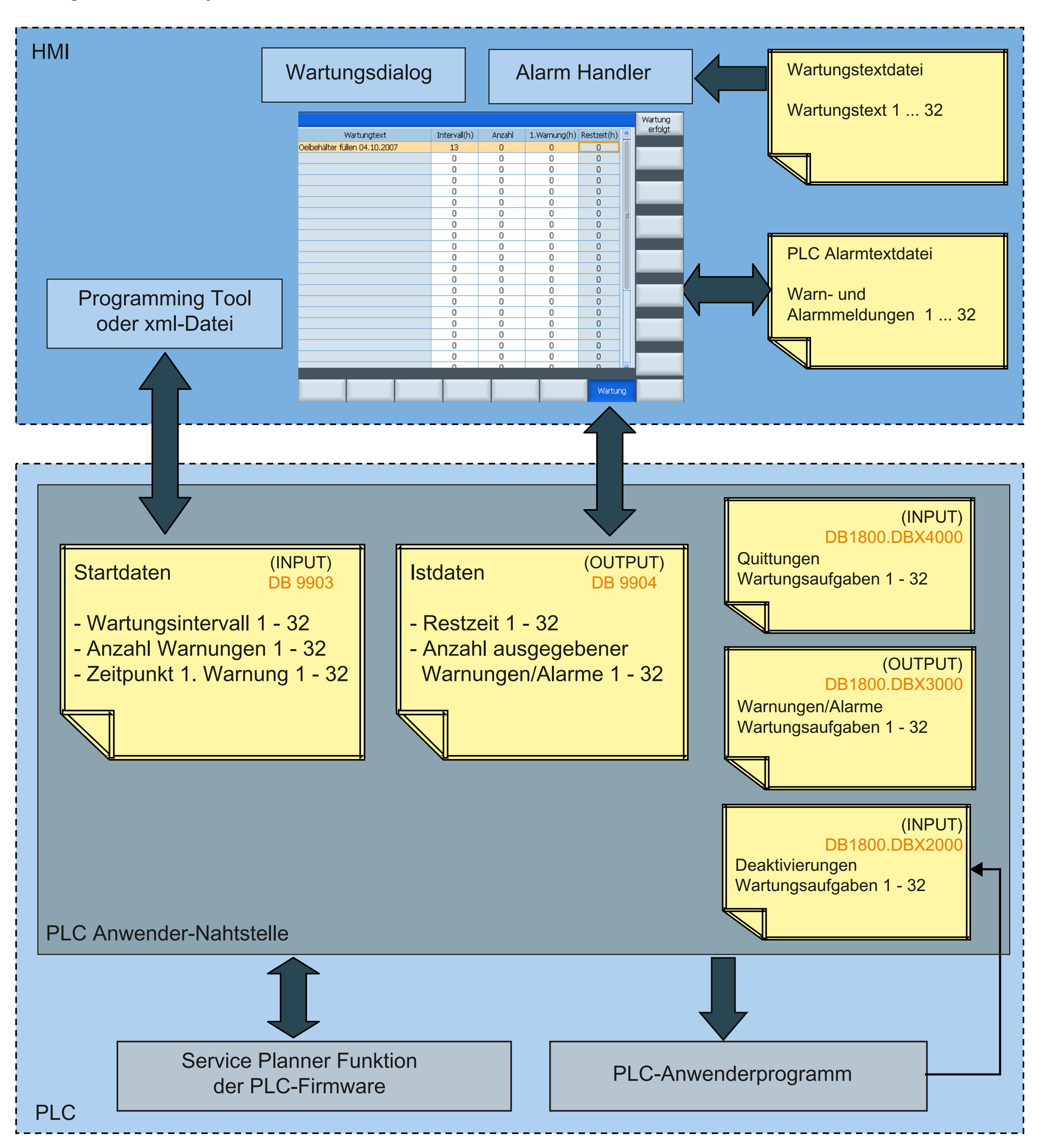


Bild 8-1 Service Planner: Konfiguration

8.1 Schnittstellen im PLC-Anwenderprogramm

PLC-Anwenderprogramm

Das PLC-Anwenderprogramm bedient die Anwenderschnittstelle. Dazu gehören insbesondere die Bitschnittstellen des DB1800 und die Auswertung der Restzeit im DB9904. Es ist Sorge zu tragen, dass bei Warnungen und Alarmen entsprechende Meldungen angezeigt werden.

Bei der Projektierung dieser Meldungen sind auch sogenannte Alarmreaktionen wählbar, so ist z. B. im Fall einer Power-OFF-Meldung die Maschine danach "gesperrt".

Die Alarmmeldungen werden nach den Festlegungen der PLC Meldungen der SINUMERIK 828D projektiert. Die Texte werden mit dem Alarmtext-Editor eingegeben. Danach stehen die Texte in der Bediensoftware zur Verfügung:

- Auswertung der Alarm- und Istdaten mit dem Ziel der Generierung von PLC-Warn- und Alarmmeldungen. In die Auswertelogik können weitere Signale einbezogen werden.
- Optionale Verknüpfung der Deaktivierungsbits mit Merkern oder I/O-Signalen.

Hinweis

Ein Beispielprogramm wird in der PLC-Funktionsbibliothek mitgeliefert. Dieses kann der Maschinenhersteller entsprechend seinen Erfordernissen adaptieren.

Schnittstellen zur PLC

Folgende Datenbereichestehen in der Anwenderschnittstelle zur Verfügung:

Datenbaustein	Bedeutung
DB9903	Initialdaten
DB9904	Istdaten
DB1800.DBB2000	Aufgaben deaktivieren
DB1800.DBB3000	Alarme
DB1800.DBB4000	Quittungen
DB1800.DBB5000	Quittungssperre

DB9903: Initialdaten

DB9903	Initialdaten-Tabelle [r16]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW0	Intervall 1 [h]							
DBW2	Zeitpunkt der ersten Warnung 1 [h]							
DBW4	Anzahl auszugebender Warnungen 1							
DBW6	reserviert 1							
DBW8	Intervall 2 [h]							

DB9903	Initialdaten-Tabelle [r16]
DBW10	Zeitpunkt der ersten Warnung 2 [h]
DBW12	Anzahl auszugebender Warnungen 2
DBW14	reserviert 2
...	...
DBW248	Intervall 32 [h]
DBW250	Zeitpunkt der ersten Warnung 32 [h]
DBW252	Anzahl auszugebender Warnungen 32
DBW254	reserviert 32

Bezeichnung	Bedeutung
Intervall	Anzahl der Stunden, nach denen die Wartung durchgeführt werden muss. Nach Ablauf dieser Zeit wird letztmalig das zur Aufgabe gehörende Warnungs-/Alarmbit gesetzt.
Zeitpunkt der ersten Warnung	Anzahl der Stunden, nach denen die erste Warnung ausgegeben wird. Dieser Zeitpunkt muss größer oder gleich dem Intervall sein.
Anzahl auszugebender Warnungen	Anzahl n der vor dem Alarm auszugebenden Warnungen. (Das Alarmbit wird also maximal (n+1)mal gesetzt, nämlich n-mal als Warnung und 1-mal als Alarm.)
reserviert	Erweiterungen vorbehalten.

Beispiel:

Intervall = 100

Zeit der 1. Warnung = 80

Anzahl auszugebender Warnungen = 2

Nach Start der Aufgabe wird nach 80 Stunden zum ersten Mal das Warnungs- / Alarmbit gesetzt, nach weiteren 10 Stunden (also nach gesamt 90 Stunden) zum zweiten Mal, und nach 100 Stunden wird letztmalig das Warnungs- / Alarmbit gesetzt.

DB9904: Istdaten

DB9904	Istdaten-Tabelle [r16]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW0	Restzeit 1 [h]							
DBW2	Anzahl ausgegebener Warnungen 1							
DBW4	reserviert_1 1							
DBW6	reserviert_2 1							
DBW8	Restzeit 2 [h]							
DBW10	Anzahl ausgegebener Warnungen 2							
DBW12	reserviert_1 2							
DBW14	reserviert_2 2							

DB9904	Istdaten-Tabelle [r16]
...	...
DBW248	Restzeit 32 [h]
DBW250	Anzahl ausgegebener Warnungen 32
DBW252	reserviert_1 32
DBW254	reserviert_2 32

Bezeichnung	Bedeutung
Restzeit	Anzahl der Stunden, die nach dem Start der Aufgabe noch bis zu deren Ablauf verbleiben. Restzeit ≠ 0 und zugehöriges Alarmbit gesetzt: Warnung Restzeit = 0 und zugehöriges Alarmbit gesetzt: Alarm
Anzahl ausgegebener Warnungen	Anzahl n der bereits ausgegebenen Warnungen. Ist das Intervall vollständig abgelaufen, beträgt der ausgegebene Wert (n+1): n = "Anzahl auszugebender Warnungen" 1 = Alarm am Intervallende
Reserviert_1, ~_2	Erweiterungen vorbehalten.

Beispiel:

Intervall = 100, Zeit der 1. Warnung = 80, Anzahl auszugebender Warnungen = 2

Nach Start der Aufgabe wird die Restzeit stündlich dekrementiert.

- Nach 80 Stunden beträgt die Restzeit 20 h und die Anzahl ausgegebener Warnungen wird von 0 auf 1 erhöht.
- nach weiteren 10 Stunden (also nach gesamt 90 Stunden) beträgt die Restzeit 10 h und die Anzahl ausgegebener Warnungen wird von 1 auf 2 erhöht.
- nach 100 Stunden ist die Restzeit 0 und die Anzahl ausgegebener Warnungen beträgt 3 (= 2 Warnungen plus 1 Alarm).

DB1800: Quittungen

DB1800	Quittungen [r/w]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB4000	Quittung 8	Quittung 7	Quittung 6	Quittung 5	Quittung 4	Quittung 3	Quittung 2	Quittung 1
DBB4001	Quittung 16	Quittung 15	Quittung 14	Quittung 13	Quittung 12	Quittung 11	Quittung 10	Quittung 9
DBB4002	Quittung 24	Quittung 23	Quittung 22	Quittung 21	Quittung 20	Quittung 19	Quittung 18	Quittung 17
DBB4003	Quittung 32	Quittung 31	Quittung 30	Quittung 29	Quittung 28	Quittung 27	Quittung 26	Quittung 25

Bezeichnung	Bedeutung
Quittung n	Das der Aufgabe n zugeordnete Quittungsbit: Unter der Voraussetzung, dass das korrespondierende Quittungssperrbit nicht gesetzt ist, startet das Setzen des Quittungsbits die Aufgabe neu, insbesondere werden die Istdaten der Aufgabe gesetzt: • Restzeit := Intervall, • Anzahl ausgegebener Warnungen := 0 Das Bit wird automatisch am Ende des PLC-Zyklus zurückgesetzt.

Beispiel:

Intervall = 100, Zeit der 1. Warnung = 80, Anzahl auszugebender Warnungen = 2

Nach Setzen des zugehörigen Quittungsbits wird die Restzeit wieder auf die Intervallzeit gesetzt und die Anzahl ausgegebener Warnungen ist Null – vorausgesetzt, das zugehörige Quittungssperrbit ist nicht gesetzt.

DB1800: Alarme

DB1800	Warnungen / Alarme [r]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB3000	Alarm 8	Alarm 7	Alarm 6	Alarm 5	Alarm 4	Alarm 3	Alarm 2	Alarm 1
DBB3001	Alarm 16	Alarm 15	Alarm 14	Alarm 13	Alarm 12	Alarm 11	Alarm 10	Alarm 9
DBB3002	Alarm 24	Alarm 23	Alarm 22	Alarm 21	Alarm 20	Alarm 19	Alarm 18	Alarm 17
DBB3003	Alarm 32	Alarm 31	Alarm 30	Alarm 29	Alarm 28	Alarmm27	Alarm 26	Alarm 25

Bezeichnung	Bedeutung
Alarm n	Das der Aufgabe n zugeordnete Alarmbit. Das Bit wird für jeweils einen PLC-Zyklus gesetzt: als Warnung (Restzeit ≠ 0) und als Alarm (Restzeit = 0).

DB1800: Aufgaben deaktivieren

DB1800	Aufgaben-Deaktivierung [r/w]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB2000	Deakti-vierung 8	Deakti-vierung 7	Deakti-vierung 6	Deakti-vierung 5	Deakti-vierung 4	Deakti-vierung 3	Deakti-vierung 2	Deakti-vierung 1
DBB2001	Deakti-vierung 16	Deakti-vierung 15	Deakti-vierung 14	Deakti-vierung 13	Deakti-vierung 12	Deakti-vierung 11	Deakti-vierung 10	Deakti-vierung 9

DB1800	Aufgaben-Deaktivierung [r/w]							
DBB2002	Deakti-vierung 24	Deakti-vierung 23	Deakti-vierung 22	Deakti-vierung 21	Deakti-vierung 20	Deakti-vierung 19	Deakti-vierung 18	Deakti-vierung 17
DBB2003	Deakti-vierung 32	Deakti-vierung 31	Deakti-vierung 30	Deakti-vierung 29	Deakti-vierung 28	Deakti-vierung 27	Deakti-vierung 26	Deakti-vierung 25

Bezeichnung	Bedeutung
Deaktivierung n	Das der Aufgabe n zugeordnete Deaktivierungsbit. Wird das Bit über das HMI oder vom PLC-Anwenderprogramm gesetzt, wird der augenblickliche Zustand der Aufgabe n eingefroren und diese nicht mehr bearbeitet. TRUE: Aufgabe deaktiviert FALSE: Aufgabe aktiv Damit ist es z. B. möglich, das Wartungsintervall an die tatsächliche Laufzeit der Baugruppen anzupassen.

DB1800: Quittungssperre

DB1800	Quittungssperre [r/w]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB5000	Quittungs-sperre 8	Quittungs-sperre 7	Quittungs-sperre 6	Quittungs-sperre 5	Quittungs-sperre 4	Quittungs-sperre 3	Quittungs-sperre 2	Quittungs-sperre 1
DBB5001	Quittungs-sperre 16	Quittungs-sperre 15	Quittungs-sperre 14	Quittungs-sperre 13	Quittungs-sperre 12	Quittungs-sperre 11	Quittungs-sperre 10	Quittungs-sperre 9
DBB5002	Quittungs-sperre 24	Quittungs-sperre 23	Quittungs-sperre 22	Quittungs-sperre 21	Quittungs-sperre 20	Quittungs-sperre 19	Quittungs-sperre 18	Quittungs-sperre 17
DBB5003	Quittungs-sperre 32	Quittungs-sperre 31	Quittungs-sperre 30	Quittungs-sperre 29	Quittungs-sperre 28	Quittungs-sperre 27	Quittungs-sperre 26	Quittungs-sperre 25

Bezeichnung	Bedeutung
Quittungssperre n	Das der Aufgabe n zugeordnete Quittungssperrbit. Wird das Bit über das HMI oder vom PLC-Anwenderprogramm gesetzt, wird trotz Setzens des Quittungsbits die Aufgabe nicht quittiert. TRUE: Quittieren der Aufgabe gesperrt FALSE: Quittieren der Aufgabe erlaubt Damit ist es z. B. möglich, einen Sensor, der die Erfüllung der Wartungsaufgabe signalisiert, in das PLC-Anwenderprogramm einzubinden und gegebenenfalls das Quittieren zu verbieten.

8.2 Schnittstellen zur Bediensoftware

Übersicht

Zum Bearbeiten von Wartungsaufgaben haben Sie folgende Möglichkeiten:

- Verwaltung der Daten im PLC-Anwenderprogramm über das Programming Tool.
- Anlegen der Wartungsaufgaben über eine XML-Schnittstelle.

Durch diese Aufteilung werden Inkonsistenzen zwischen dem PLC-Baustein und dem XML-Skript vermieden, wenn der PLC-Baustein im PLC-Anwenderprogramm über das Programming Tool oder durch ein Inbetriebnahmearchiv verändert wird.

Projektierung

Es können maximal 32 Wartungsaufgaben projektiert werden. Im Projektierungsmodus werden folgende Spalten angezeigt:

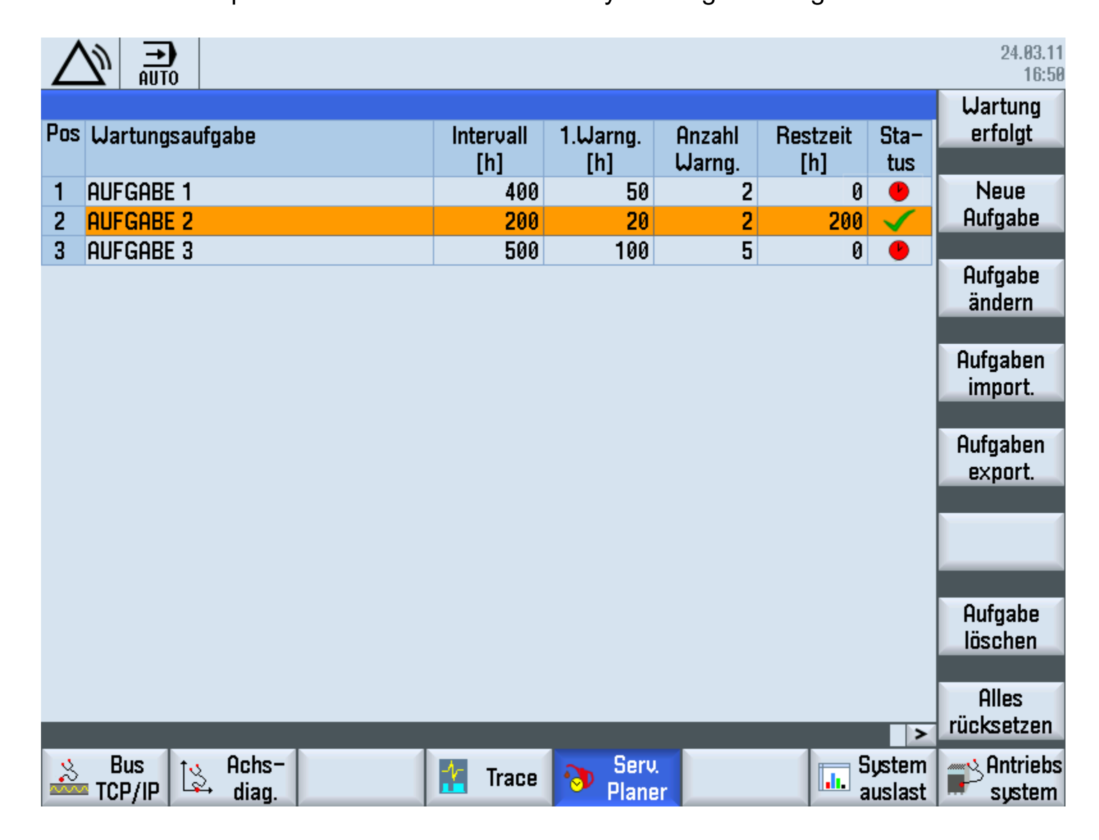
Bezeichnung der Spalte	Bedeutung
Wartungsaufgabe	Bezeichnung der Wartungsaufgabe
Intervall [h]	Maximale Zeit bis zur nächsten Wartung in Stunden; ist dieser Wert ≠ 0, wird dieser Datensatz als gültige Wartungsaufgabe von der PLC akzeptiert.
1.Warnung [h]	Zeit in Stunden, bei der erstmalig eine Warnung angezeigt wird; dieser wert muss kleiner als der des Intervalls sein.
Anzahl Warnungen	Anzahl der Warnungen, die von der PLC ausgegeben werden, bevor die PLC nach Ablauf des Intervalls (Restzeit == 0) letztmalig das Alarmbit setzt.
Restzeit [h]	Zeit bis zum Ablauf des Intervalls in Stunden
Status	• Ein grüner Haken zeigt an, dass die Zeit bis zur nächsten Wartung noch läuft. • Eine rote Uhr zeigt an, dass eine Wartungsaufgabe ansteht.

Der Dialog wird abhängig von der Zugriffsstufe mit unterschiedlichem Inhalt aufgerufen.

Projektierungsmodus

Zugriffsstufe 2: Service

In diesem Modus können Wartungsaufgaben angelegt, geändert und auch gelöscht werden. Zusätzlich können die Wartungsaufgaben quittiert werden. Alle Spalten sind sichtbar. Zwischen den Spalten wird mit <Tab> oder <Key Left/Right> navigiert.



Hinweis

Nummernzuordnung

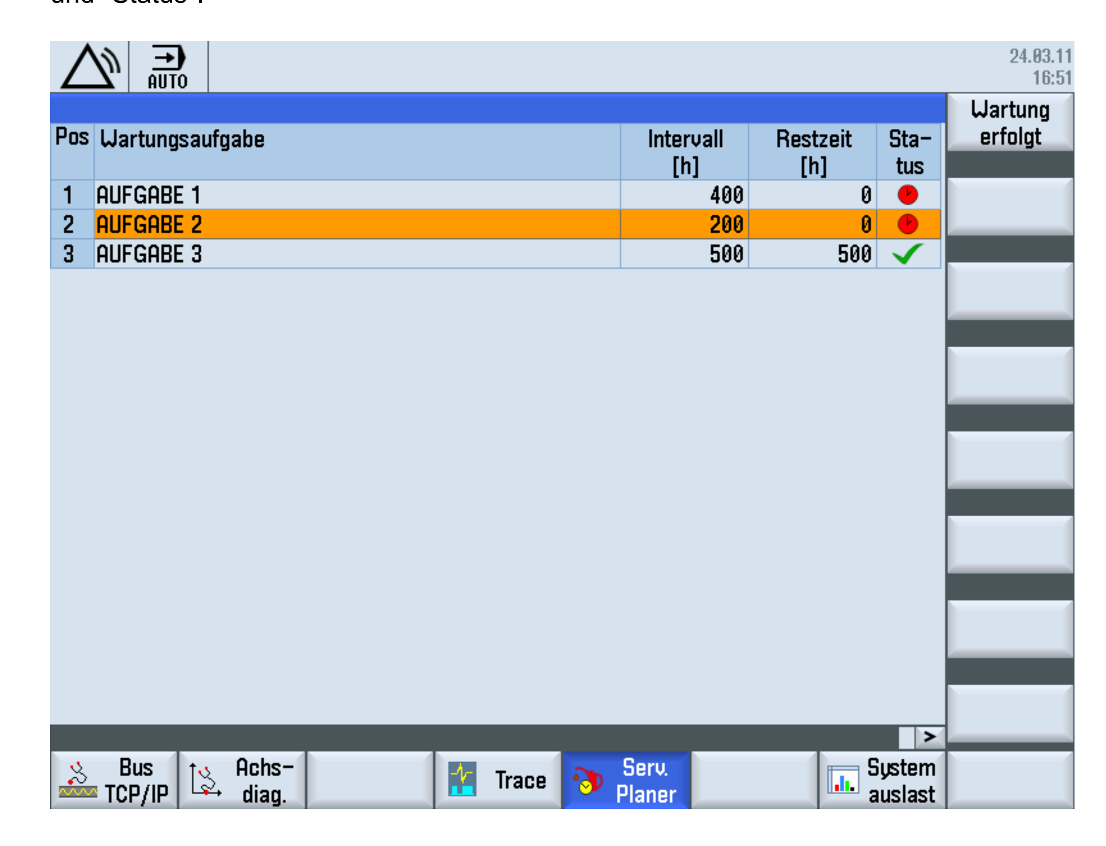
Beim Anlegen von Aufgaben erfolgt eine automatische Nummernzuordnung. Wird dies nicht gewünscht, müssen die Aufgaben mit den betreffenden Nummern projektiert werden.

Dies empfiehlt sich zum Beispiel beim versehentlichen Löschen der Aufgabe m, wenn diese wegen der Auswertung im PLC-Anwenderprogramm wieder unter derselben Nummer angelegt werden muss.

Standardmodus

Zugriffsstufe 3: Anwender

Im Standardmodus wird der aktuelle Zustand der Wartungsaufgaben angezeigt. Es sind folgende Spalten sichtbar, aber nicht editierbar: "Wartungsaufgabe", "Intervall", "Restzeit" und "Status".



8.3 So importieren und exportieren Sie Wartungsaufgaben

Übersicht

Im Dialog stehen zwei Softkeys zur Verfügung, um die Konfigurationsdateien mit den Wartungsaufgaben einzulesen und auszulesen:

- Wartungsaufgaben importieren
- Wartungsaufgaben exportieren

Wartungsaufgaben projektieren

Vorgehensweise:

1. Erstellen Sie eine Konfigurationsdatei nach dem unten beschriebenen Aufbau.

   Der Name der Datei in Kleinbuchstaben ist beliebig wählbar.
2. Erstellen Sie mindestens eine Datei oem_maintenance_<lng>.ts, die die sprachabhängigen Bezeichnungen für die Wartungsaufgaben enthält.
3. Kopieren Sie die beiden Dateien auf ein Speichermedium (CompactFlash Card oder USB-FlashDrive).
4. Stecken Sie das Speichermedium in den entsprechenden Steckplatz an der Frontseite.
5. Betätigen Sie den Softkey "Wartungsaufgaben importieren".
6. Bestätigen Sie mit "OK".

   Die Dateien werden kopiert. Die sprachabhängigen Texte sind sofort sichtbar.

Um Änderungen oder Ergänzungen vorzunehmen, betätigen Sie den Softkey "Wartungsaufgaben exportieren". Damit Kopieren Sie die Dateien auf ein Speichermedium, um Sie auf einem externen Editor zu bearbeiten.

Aufbau der xml-Konfigurationsdatei

Folgende Bezeichner sind zulässig:

Bezeichner	Bedeutung
<MAINTENANCE_TASK>	Main Tag
<TASK_ID>	Nummer der Wartungsaufgabe
<INTERVALL>	Wartungsintervallzeit
<FIRST_WARNING>	Zeitintervall bis die erste Warnung ausgegeben wird.
<NUMBER_OF_WARNING>	Anzahl der Warnungen

Der Name der Datei ist beliebig wählbar, zum Beispiel task.xml.

task.xml

```
<MAINTENANCE>
  <MAINTENANCE_TASK>
    <TASK_ID>0</TASK_ID>
    <INTERVALL>3</INTERVALL>
    <FIRST_WARNING>1</FIRST_WARNING>
    <NUMBER_OF_WARNING>1</NUMBER_OF_WARNING>
  </MAINTENANCE_TASK>
  <MAINTENANCE_TASK>
    <TASK_ID>1</TASK_ID>
    <INTERVALL>3</INTERVALL>
    <FIRST_WARNING>2</FIRST_WARNING>
    <NUMBER_OF_WARNING>1</NUMBER_OF_WARNING>
  </MAINTENANCE_TASK>
  ...
  <MAINTENANCE_TASK>
    <TASK_ID>2</TASK_ID>
    <INTERVALL>3</INTERVALL>
    <FIRST_WARNING>2</FIRST_WARNING>
    <NUMBER_OF_WARNING>1</NUMBER_OF_WARNING>
  </MAINTENANCE_TASK
</MAINTENANCE>
```

Aufbau der Datei oem_maintenance_<lng>.ts

Diese Datei hat die Erweiterung ".ts" und enthält alle sprachabhängigen Wartungstexte, die im Dialog eingegeben wurden. Nach dem nächsten Hochlauf des Systems steht diese Datei im Binärformat (*.qm) zur Verfügung.

Zielverzeichnis: `/oem/sinumerik/hmi/lng`

oem_maintenance_<lng>.ts

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE TS>
<TS>
  <context>
    <name>maintenance</name>
    <message>
      <source>1</source>
      <translation>Wartungsaufgabe 1</translation>
      <chars>44</chars>
    </message>
```

oem_maintenance_<lng>.ts

```
    <message>
      <source>2</source>
      <translation>Wartungsaufgabe 2</translation>
      <chars>44</chars>
    </message>
. . .
    <message>
      <source>32</source>
      <translation>Wartungsaufgabe 32</translation>
      <chars>44</chars>
    </message>
  </context>
</TS>
```

Texteinträge editieren

Die Wartungstexte werden im Dialog zusammen mit den Werten für Intervall, Zeit der ersten Warnung und Warnungsanzahl eingetragen. Zusätzlich kann die *.ts-Datei im Alarmtexteditor bearbeitet werden, wenn in der Konfigurationsdatei "oem_alarms_config.xml" der notwendige Eintrag vorhanden ist.

Einbindung in das vorhandene Sprachkonzept

Der Service Planner liest beim Start die Datei oem_maintenance_<lng>.ts mit der im Sprachauswahlmenü eingestellten Sprache. Ist diese nicht vorhanden, wird die englische Version gelesen, die bei der Inbetriebnahme vorhanden sein muss.

8.4 Wartungsaufgaben quittieren

Wartungsaufgaben quittieren

Nach Abschluss der Wartungstätigkeiten werden die Wartungsaufgaben durch das PLC-Anwenderprogramm oder im Dialog der Bediensoftware mit dem Softkey "Wartung erfolgt" quittiert.

Durch das Quittieren im Dialog wird das zur Wartungsaufgabe gehörende Quittungsbit gesetzt, die PLC löscht in den Ist-Daten die "Anzahl ausgegebener Warnungen/Alarme" und lädt die Restzeit mit dem Intervallwert. Daran erkennt der Anwender das erfolgreiche Quittieren der Wartungsaufgabe.

Der Quittungszeitpunkteiner Wartungsaufgabe hat folgende Wirkung:

- **Quittung vor Ablauf des Intervalls**

  Das Quittieren des Wartungsintervalls ist jederzeit möglich. Vorfristiges Quittieren bedeutet den vorfristigen Start eines neuen Wartungsintervalls.

- **Quittung nach Ablauf des Intervalls**

  Das Quittieren des Wartungsintervalls bewirkt den Neustart der Aufgabe.

Hinweis

Zugriffsstufe zum Quittieren

Die Zugriffsstufe zum Quittieren einer Wartungsaufgabe wird durch folgendes Maschinendatum bestimmt: MD51235 $MNS_ACCESS_RESET_SERV_PLANNER

Voreinstellung: Zugriffsstufe 2 "Service"

Easy Extend 9

9.1 Funktionsübersicht

Zielsetzung

Zusatzgeräte lassen sich auf einfache Art und Weise mit "Easy Extend" in Betrieb nehmen, aktivieren, deaktivieren oder testen. Die verfügbaren Geräte und Gerätezustände zeigt die Steuerung in einer Liste an. Das System kann maximal 64 Geräte verwalten.

Das Aktivieren oder Deaktivieren eines Gerätes erfolgt per Softkey-Bedienung.

Die Funktion "Easy Extend" steht im Bedienbereich "Parameter" → Erweiterungsleiste → "Easy Extend" zur Verfügung.

Projektierung

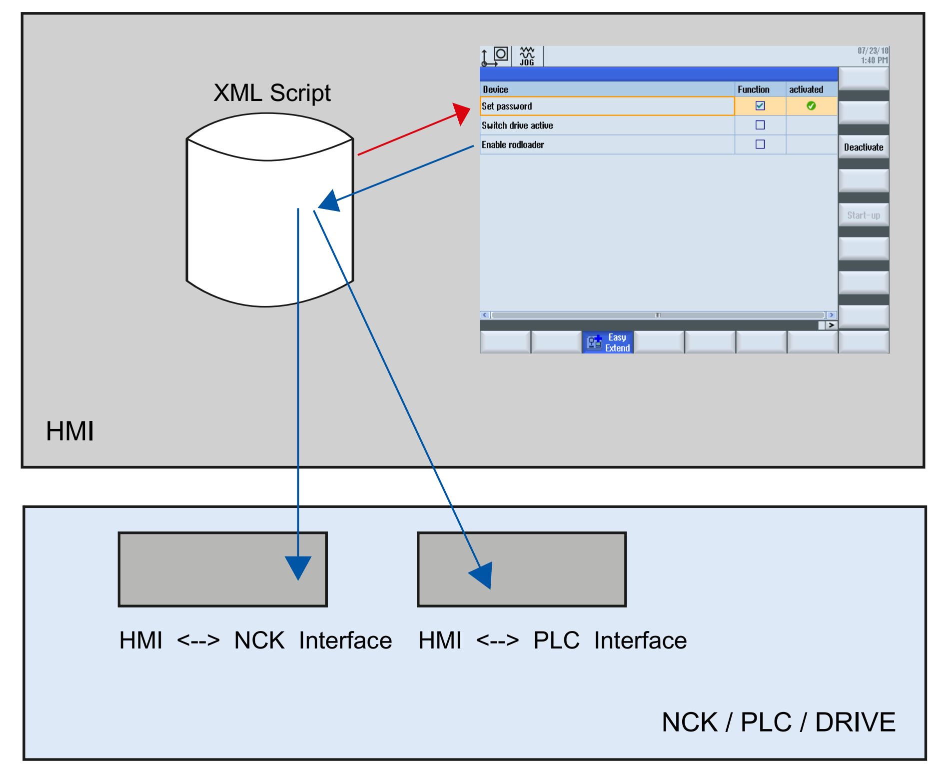


Bild 9-1 Funktionsweise von Easy Extend

Um "Easy Extend" nutzen zu können, sind vom Maschinenhersteller folgende Funktionen zu projektieren:

- **Schnittstelle PLC ↔ HMI**

  Die Verwaltung der optionalen Geräte wird über die Schnittstelle zwischen der Bedienoberfläche und der PLC abgewickelt.

- **Skript-Verarbeitung**

  Der Maschinenhersteller hinterlegt in einem Anweisungsskript die Abläufe, die zum Installieren, Aktivieren, Deaktivieren und Testen eines Gerätes auszuführen sind.

- **Parameter-Dialog (optional)**

  Der Parameter-Dialog dient zum Anzeigen von Geräte-Informationen, die in der Skript-Datei hinterlegt sind.

Ablage der Dateien

Die Ablage der zu Easy Extend gehörenden Dateien erfolgt auf der System CompactFlash Card im Verzeichnis "oem" (MANUFACTURER) und "oem_i" (INDIVIDUAL).

Datei	Name	Zielverzeichnis
Textdatei	oem_aggregate_xxx.ts	/oem/sinumerik/hmi/lng/ /oem_i/sinumerik/hmi/lng/
Skriptdatei	agm.xml	/oem/sinumerik/hmi/dvm /oem_i/sinumerik/hmi/dvm
Archivdatei	beliebig	/oem/sinumerik/hmi/dvm/archives /oem_i/sinumerik/hmi/dvm/archives
PLC-Anwenderprogramm	beliebig	PLC

9.2 Projektierung im PLC-Anwenderprogramm

Konfigurationen laden

Die erstellten Konfigurationen werden zusammen mit der Skript- und Textdatei in das Herstellerverzeichnis der Steuerung übertragen. Zusätzlich ist das entsprechende PLC-Anwenderprogramm zu laden.

Programmierung der Geräte

Die Kommunikation zwischen der Bedienkomponente und der PLC erfolgt im PLC-Anwenderprogramm über den **Datenbaustein DB9905**, bei dem 128 Wörter für die Verwaltung von maximal 64 Geräten reserviert sind.

Für jedes Gerät werden vier Bytes mit folgender Bedeutung verwendet:

Byte	Bit	Beschreibung	
0	0	== 1	Gerät ist in Betrieb genommen (HMI Rückmeldung)
	1	== 1	Gerät ist zu aktivieren (HMI Anforderung)
	2	== 1	Gerät ist zu deaktivieren (HMI Anforderung)
	3-7	reserviert	
1	0-7	reserviert	
2	0	== 1	Gerät ist aktiv (PLC Rückmeldung)
	1	== 1	Gerät ist fehlerhaft
	2-7	reserviert	
3	0-7	eindeutige Kennzeichnung des Gerätes	

Achsen hinzufügen

Wird die Maschine um Maschinenachsen erweitert, ist eine feste Reihenfolge beim Anbau der Drive-Objekte (DO) einzuhalten, da das Inbetriebnahmearchiv die Konstellation der Referenzmaschine des Maschinenherstellers beinhaltet und nicht bei einer geänderten Reihenfolge angewendet werden kann.

Es wird empfohlen für die "Steuerungskomponenten" folgende Einstellungen zu wählen:

- NC-Daten
- PLC-Daten
- Antriebs-Daten
  - ACX-Format (binär)

Hinweis

Inbetriebnahmearchive

- Um Inbetriebnahmearchive im Easy Extend-Skript zu nutzen, sind diese Archive **ohne** HMI-Daten zu erstellen!

Änderung der Maschinenkonfiguration

- Ist das Ändern von Antriebsmaschinendaten notwendig, sollten diese zuerst in der Steuerung angepasst werden. Dieser Vorgang ist für alle Geräte und Konstellationen zu wiederholen.

Siehe auch

So erzeugen Sie ein Inbetriebnahmearchiv (Seite 392)

9.3 Optionsbits für Maschinenhersteller und Händler

Aufteilung der Optionsbits

Die Optionsbits werden in Blöcken von 16 Bits verwaltet. Für den Händler ist gemäß Voreinstellung Block 3 reserviert. Damit stehen dem Maschinenhersteller 48 Optionen und dem Händler 16 Optionen zur Verfügung.

Zur Optionsverwaltung verwendet die Funktion Easy Extend folgende Maschinendaten:

MD14510 $MN_USER_DATA_INT[0] bis MD14510 $MN_USER_DATA_INT[3]

Diese Zuordnung der Adressen kann im Skript mit dem Bezeichner OPTION_MD auch so umdefiniert werden, dass der Händler einen eigenen Adressbereich angeben kann. Möchte der Händler mehr als 16 Optionen für einen Maschinentyp anbieten, sind die nicht verwendeten Blöcke zu ermitteln. Dazu ist eine Absprache mit dem Maschinenhersteller notwendig.

Um den Händlerbereich bekanntzugeben, ist der Block 3 per Anweisung in das Skript zu programmieren. Weiterhin sollte für jedes Gerät ein fester Geräte-Index vergeben werden.

Beispiel, um einen beliebigen Bereich neu zu definieren:

```
<option_md name="address identifier of the data" index="<index>"/>
  index - identifier of the area index:
    0: device 0 to 16 (default setting)
    1: device 17 to 32
    2: device 33 to 48
    3: device 49 to 64
```

Gerät auf die PLC–Schnittstelle abbilden

Um eine feste Zuordnung der Geräte zum Optionsbit und zur PLC-Nahtstelle zu erreichen, ist jedem Gerät ein eindeutiger Index zu zuweisen. Die feste Zuordnung der Bereiche erfolgt mit dem Attribut option_bit. Erfolgt keine Zuweisung, sind das Bit und die Nahtstelle durch den Index bestimmt, der dem Gerät in folgender Liste zugeordnet ist:

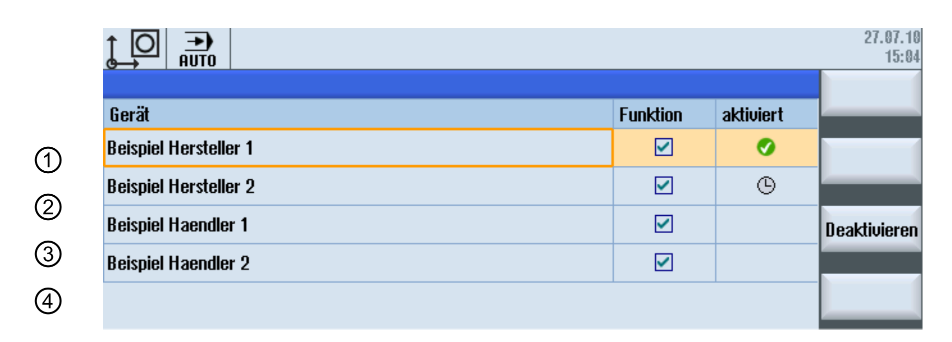
Index	MD14510	Datenbaustein	Gerätebezeichnung
0	$MN_USER_DATA_INT[0] Bit 0	DB9905.DBB0	Gerät 1
1	$MN_USER_DATA_INT[0] Bit 1	DB9905.DBB4	Gerät 2
2	$MN_USER_DATA_INT[0] Bit 2	DB9905.DBB8	Gerät 3
3	$MN_USER_DATA_INT[0] Bit 3	DB9905.DBB12	Gerät 4
....	...	...	...
47	$MN_USER_DATA_INT[2] Bit 15	DB9905.DBB188	Gerät 48
48	$MN_USER_DATA_INT[3] Bit 0	DB9905.DBB192	Gerät 49
49	$MN_USER_DATA_INT[3] Bit 1	DB9905.DBB196	Gerät 50

Index	MD14510	Datenbaustein	Gerätebezeichnung
....	...	...	...
62	$MN_USER_DATA_INT[3] Bit 14	DB9905.DBB248	Gerät 63
63	$MN_USER_DATA_INT[3] Bit 15	DB9905.DBB252	Gerät 64

Beispiel für eine feste Zuordnung:

```
<device option_bit="bit number">
...
</device>
```

Beispiel



Blockweise Zuordnung:

①	DB9905.DBB0	Gerät 1
②	DB9905.DBB4	Gerät 2
	...	
③	DB9905.DBB192	Gerät 49
④	DB9905.DBB196	Gerät 50

Zuordnung auf freie Plätze:

①	DB9905.DBB0	Gerät 1
②	DB9905.DBB4	Gerät 2
	...	
③	DB9905.DBB72	Gerät 19
④	DB9905.DBB76	Gerät 20
	...	

Rückmeldung von PLC

Kann der Händler keine eigene PLC-Logik integrieren, müssen die Rückmeldesignale der PLC durch das Skript gesetzt werden.

Siehe auch

Attribut option_bit: Aufbau des XML-Skripts (Seite 258)

XML-Bezeichner: OPTION_MD (Seite 271)

9.4 Darstellung auf der Bedienoberfläche

Dialoge auf der Bedienoberfläche

Folgende Dialoge sind für Easy Extend verfügbar:

- Die Steuerung bietet einen **projektierbaren Dialog** an, in dem die verfügbaren Geräte angezeigt werden.
- Wurde noch keine Erstinbetriebnahme durchgeführt, öffnet die Steuerung den **Dialog zur Inbetriebnahme**.

Ist für das Gerät eine Inbetriebnahmeprozedur (XML-Anweisung: "START_UP") programmiert und das Gerät noch nicht in Betrieb genommen, startet die Steuerung die Inbetriebnahmeprozedur.

Dazu erfolgt zuerst eine vollständige Datensicherung, bevor die in der Skriptdatei hinterlegten Inbetriebnahmearchive eingelesen werden. Als Archivtypen sind Standard- oder Datenklassenarchive zulässig: *.arc und *.ard

- Im Fehlerfall kann der Inbetriebsetzer entscheiden, ob das Zurückrollen der Inbetriebnahme erfolgen soll oder er manuell eventuelle Fehler in den Maschinenkonfigurationen beheben möchte.
- Mit der Funktion "Abbruch" ist ein vorzeitiger Abbruch der Inbetriebnahme möglich. Danach kopiert die Steuerung die zuvor gesicherten Inbetriebnahmedateien zurück.

Ist nach erfolgreichem Abschluss der Inbetriebnahme das Ausschalten der Maschine erforderlich, kann über die XML-Anweisung "POWER_OFF" programmiert werden, dass eine entsprechende Meldung an der Steuerung ausgegeben wird.

9.5 Sprachabhängigen Text erstellen

Ersetzungszeichen für sprachabhägigen Text

Das System bietet die Möglichkeit, CONTROL-Eigenschaften (Attributwerte) zur Laufzeit festzulegen. Um diese Funktion nutzen zu können, ist die gewünschte Eigenschaft in einer lokalen Variablen bereitzustellen und der Variablenname mit vorangestellten $-Zeichen als Attributwert dem Tag zu übergeben.

Beispiel:

```
<let name="caption_device_variable" type="string">my_device 3</let>

  <DEVICE option_bit="8">
    <!-- Direkt: Es wird der eingegebene Ausdruck angezeigt. -->
    <NAME>my device 1</NAME>
  </DEVICE>

  <DEVICE option_bit="10">
    <!-- Indirekt: Es wird der für den Textbezeichner caption_id_my_device2
hinterlegte Text angezeigt. -->
    <NAME>$$caption_id_my_device2</NAME>
    <form>
      <init>
        <!-- Indirekt: Es wird der für den Textbezeichner
form_caption_device2 hinterlegte Text angezeigt. -->
        <caption>$$form_caption_device2</caption>
      </init>
    </form>
  </DEVICE>
  <!-- Indirekt: Es wird der Inhalt der Variablen caption_device_variable angezeigt.
-->

  <DEVICE option_bit="11">
    <NAME>$$$caption_device_variable</NAME>
  </DEVICE>
```

Aufbau der Textdatei

Die xml-Dateien mit den sprachabhängigen Texten sind im UTF8-Format zu erstellen:

Beispiel: oem_aggregate_eng.ts

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE TS>
<TS>
<context>
<name>EASY_EXTEND</name>
  <message>
    <source>form_caption_id_my_device2</source>
    <translation>my_device2</translation>
    <chars>64</chars>
  </message>
  <message>
    <source>form_caption_device2</source>
    <translation>my_input_form</translation>
    <chars>64</chars>
  </message>
</context>
</TS>
```

9.6 Beispiele

9.6.1 Beispiel mit Steuerungselementen

Beispiel für ein Kombinationslistenfeld

Wird als Feldtyp "combobox" ausgewählt, sind zusätzlich die darzustellenden Ausdrücke festzulegen. Dafür ist der Bezeichner <item> zu verwenden. Das Kombinationslistenfeld speichert den Index des aktuell ausgewählten Textes in der zum "CONTROL" gehörenden Variablen (Variablenname). Der Index beginnt mit dem Wert 1.

```
<control name = "button1" xpos = "10" ypos = "10" fieldtype = "combobox">
<item>text1</item>
<item>text2</item>
<item>text3</item>
<item>text4</item>
</control>
```

Beispiel für eine Wertzuweisung

Soll ein beliebiger Integer – Wert einem Ausdruck zugeordnet werden, ist das Attribut value="Wert" dem Tag hinzuzufügen. Anstelle der fortlaufenden Nummerierung enthält jetzt die Control – Variable den zugewiesenen Wert des Items.

```
<control name = "button1" xpos = "10" ypos = "10" fieldtype = "combobox">
<item value = "10" >text1</item>
<item value = "20" >text2</item>
<item value = "12" >text3</item>
<item value = "1" >text4</item>

</control>
```

Hinweis

Attribut "hotlink"

Das Attribut hotlink führt zu einem zyklischen Aktualisieren des entsprechenden Controls. Das bedeutet, wird ein Wert eingegeben, überschreibt der nachfolgende Aktualisierungszyklus den eingegebenen Wert. Um diesen Verhalten zu umgehen, ist mittels DATA_ACCESS – Tag das sofortige Speichern der Eingaben zu aktivieren.

Eine weitere Möglichkeit besteht darin, das Tag SOFTKEY_OK in die Form aufzunehmen. Dieser Tag wird vor dem Schließen des Dialoges ausgeführt. In diesem Block kann mit der Anweisung UPDATE_CONTROLS der Datenabgleich zwischen den Control- und Referenzvariablen durchgeführt werden.

9.6.2 Beispiel mit Parametern zur Unterstützung der Inbetriebnahme

Dialog mit Zusatzparametern

Die Eingabefelder listen ausgewählte Antriebsparameter auf.

```
<DEVICE>
  <list_id>3</list_id>
  <name> "Test form" </name>
  <form>

  <init>
    <caption>Equipment Manager</caption>
    <control name = "edit1" xpos = "400" ypos = "34" refvar = "drive/dc/p105[DO5]"
/>
    <control name = "edit1" xpos = "400" ypos = "54" refvar =
"$MC_AXCONF_MACHAX_USED[4]" />
    <control name = "edit1" xpos = "400" ypos = "74" refvar = "drive/dc/p971[DO5]"
/>
    <control name = "edit1" xpos = "400" ypos = "94" refvar = "drive/dc/r2[DO5]" />
  </init>

  <paint>
  <text xpos = "40" ypos = "34">dc[DO5]/p105</text>
  <text xpos = "40" ypos = "54">$MC_AXCONF_MACHAX_USED[4]</text>
  <text xpos = "40" ypos = "74">dc[DO5]/p971</text>
  <text xpos = "40" ypos = "94">dc[DO5]/r2</text>
  </paint>
  </form>
```

```
</DEVICE>
```

Dialog mit Kombinationslistenfeld

```
<form>

<init>
<caption>selected machine data</caption>
<DATA_ACCESS type="true" />
<!-- switch on the direct access to the NC variables -->
<control name = "edit1" xpos = "322" ypos = "34"
refvar="$MN_AXCONF_MACHAX_NAME_TAB[0]" />
<control name = "edit2" xpos = "322" ypos = "54"
refvar="$MN_AXCONF_MACHAX_NAME_TAB[1]" />
<control name = "edit3" xpos = "322" ypos = "74"
refvar="$MN_AXCONF_MACHAX_NAME_TAB[2]" />
<control name = "edit4" xpos = "322" ypos = "94"
refvar="$MN_AXCONF_MACHAX_NAME_TAB[3]" />

<control name = "edit5" xpos = "322" ypos = "114" refvar="$MA_IS_ROT_AX[AX1]"
hotlink="true" />
<control name = "edit6" xpos = "322" ypos = "134" refvar="$MA_IS_ROT_AX[AX2]"
hotlink="true" />
<control name = "edit7" xpos = "322" ypos = "154" refvar="$MA_IS_ROT_AX[AX3]"
hotlink="true" />

<!-- using the control type combo box to display the rotation axis value -->
<control name = "edit5" xpos = "322" ypos = "194" refvar="$MA_IS_ROT_AX[AX1]"
fieldtype = "combobox" hotlink="true" >
<item value= "0" >no</item>
<item value= "1" >yes</item>
</control>

<control name = "edit6" xpos = "322" ypos = "214" refvar="$MA_IS_ROT_AX[AX2]"
fieldtype = "combobox" hotlink="true" >
<item value= "0" >no</item>
<item value= "1" >yes</item>
</control>

<control name = "edit7" xpos = "322" ypos = "234" refvar="$MA_IS_ROT_AX[AX3]"
fieldtype = "combobox" hotlink="true" >
<item value= "0" >no</item>
```

```
<item value= "1" >yes</item>
</control>

</init>

<paint>
<text xpos = "23" ypos = "34">AXCONF_MACHAX_TAB[0]</text>
<text xpos = "23" ypos = "54">AXCONF_MACHAX_TAB[1]</text>
<text xpos = "23" ypos = "74">AXCONF_MACHAX_TAB[2]</text>
<text xpos = "23" ypos = "94">AXCONF_MACHAX_TAB[3]</text>
<text xpos = "23" ypos = "114">Is rot axis 1</text>
<text xpos = "23" ypos = "134">Is rot axis 2</text>
<text xpos = "23" ypos = "154">Is rot axis 3</text>

<text xpos = "23" ypos = "174">using combobox control</text>

<text xpos = "23" ypos = "194">Is rot axis 1</text>
<text xpos = "23" ypos = "214">Is rot axis 2</text>
<text xpos = "23" ypos = "234">Is rot axis 3</text>

</paint>

</form>
```

9.6.3 Anwenderbeispiel für ein Leistungsteil

Drive-Objekt aktivieren

Das zu aktivierende Drive-Objekt wurde bereits in Betrieb genommen und vom Maschinenhersteller wieder deaktiviert, um die Achse(n) optional zu vermarkten.

Folgende Schritte sind zum Aktivieren der Achse auszuführen:

- Antriebsobjekt über p0105 aktivieren.
- 2. Achse in den Kanal-Maschinendaten frei schalten.
- Antriebsmaschinendaten über p0971 sichern.
- Warten, bis die Daten geschrieben wurden.
- Neustart von NCK und Antreiben anstoßen.

Programmierung:

```
<DEVICE>
  <list_id>1</list_id>
  <name> "Antrieb aktiv schalten" </name>

  <SET_ACTIVE>
    <data name = "drive/dc/p105[DO5]">1</data>
    <data name = "$MC_AXCONF_MACHAX_USED[4]">5</data>
    <data name = "drive/dc/p971[DO5]">1</data>
    <while>
      <condition> "drive/dc/p971[DO5]" !=0 </condition>
    <while>
    <control_reset resetnc ="true" resetdrive = "true"/>
  </SET_ACTIVE>

  <SET_INACTIVE>
    <data name = "drive/dc/p105[DO5]">0</data>
    <data name = "$MC_AXCONF_MACHAX_USED[4]">0</data>
    <data name = "drive/dc/p971[DO5]">1</data>
    </while>
      <condition> "drive/dc/p971[DO5]" !=0 </condition>
    </while>
    <control_reset resetnc ="true" resetdrive = "true"/>
  </SET_INACTIVE>

</DEVICE>
```

PLC-gesteuertes Gerät aktivieren

Das Gerät wird über Merker 100.0 durch das Anwenderprogramm aktiviert und deaktiviert.

Programmierung:

```
<AGM>
<OPTION_MD NAME= "$MN_USER_DATA_INT[0]" />
  <DEVICE>
    <LIST_ID>1</LIST_ID>
    <NAME> "PLC Device" </NAME>
    <SET_ACTIVE>
    <SET_ACTIVE>
      <DATA NAME= "PLC/M100.0"> 1 </DATA>
    </SET_ACTIVE>
    <SET_INACTIVE>
      <DATA NAME= "PLC/M100.0"> 0 </DATA>
    </SET_INACTIVE>
  </DEVICE>
</AGM>
```

9.7 Beschreibung der Skript-Sprache

Skript-Sprache: erweitertes XML

Als Skript-Sprache kommt XML (Extended Markup Language) zum Einsatz, die um Datenverarbeitungs- und Hochsprachenelemente erweitert wurde.

Damit bietet die Sprache gegenüber dem Standard-XML folgende zusätzliche Eigenschaften:

- Datenablage für NC/PLC-, Inbetriebnahmedaten
- Lesen und Schreiben von NC-/PLC- und Antriebsdaten
- Ausführen von bedingten Sprüngen innerhalb eines XML-Blocks
- Abarbeiten von Programmschleifen
- Ausführen arithmetischer Operationen
- Anlegen lokaler Variablen
- Einlesen/Erzeugen eines Inbetriebnahmearchivs
- Anzeigen von Nachrichten

Zusätzlich können Skript-Elemente des SinuCom Installer mit der "up"-Ausführungsanweisung verarbeitet werden.

Programmteile des Skripts

Das Skript ist in folgende Bereiche aufgeteilt:

- Bezeichner für Easy Extend
- Bezeichner für das Gerät
- Bezeichner zur Inbetriebnahme des Geräts
- Bezeichner zum Aktivieren des Geräts
- Bezeichner zum Deaktivieren des Geräts
- Bezeichner zum Testen des Geräts
- Bezeichner für Maschinendaten und Hochsprachenelemente
- Bezeichner für den Parameter-Dialog

Die einzelnen Bezeichner sind in den nachfolgenden Kapiteln beschrieben.

9.7.1 Sonderzeichen und Operatoren

Darstellung von Sonderzeichen

Zeichen, die in der XML Syntax eine besondere Bedeutung haben, müssen umschrieben werden, damit sie von einem allgemeinen XML Interpreter korrekt dargestellt werden.

Davon sind folgende Zeichen betroffen:

Zeichen	Notation in XML	Bedeutung
<	>	größer als
>	<	kleiner als
&	&	--
"	"	Anführungszeichen (gerade)
'	'	Apostroph (Auslassungszeichen)

Zulässige Operatoren

Die Operationsanweisung verarbeitet folgende Operationen:

Operator	Notation in XML	Bedeutung
=	=	Zuweisung
==	==	gleich
!	!	nicht
!=	!=	ungleich
>	>, >	größer als
<	<, <	kleiner als
>=	>=, >=	größer gleich
<=	<=, <=	kleiner gleich
|	|	bitweise ODER-Verknüpfung
||	||	logische ODER-Verknüpfung
&	&	logische oder bitweise UND-Verknüpfung
&&	&&	logische UND-Verknüpfung
+	+	Addition
-	-	Subtraktion
*	*	Multiplikation
/	/	Division

9.7.2 Aufbau des XML-Skripts

Übersicht

Für die Beschreibung des Geräts sind folgende Bezeichner verfügbar:

- Bezeichner für Easy Extend
- Bezeichner für das Gerät
- Bezeichner zur Inbetriebnahme des Geräts

- Bezeichner zum Aktivieren des Geräts
- Bezeichner zum Deaktivieren des Geräts
- Bezeichner zum Testen des Geräts

Beschreibung

Bezeichner <tag>	Bedeutung
AGM	Bezeichner für Easy Extend
DEVICE Attribut: option_bit	Bezeichner zur Beschreibung des Geräts. Für die Optionsverwaltung wird dem Gerät eine feste Bitnummer zugewiesen.
NAME	Der Bezeichner legt den im Dialog anzuzeigenden Namen des Geräts fest. Wird eine Textreferenz verwendet, zeigt der Dialog den für den Bezeichner hinterlegten Text an.
START_UP	Der Bezeichner beinhaltet die Beschreibung der Abläufe, die zur Inbetriebnahme des Gerätes notwendig sind.
SET_ACTIVE	Der Bezeichner beinhaltet die Beschreibung der Abläufe, die zum Aktivieren des Gerätes notwendig sind.
SET_INACTIVE	Der Bezeichner beinhaltet die Beschreibung der Abläufe, die zum Abschalten des Gerätes notwendig sind.
TEST	Der Bezeichner beinhaltet die Anweisungen, mit denen ein Gerät auf dessen Funktionsfähigkeit getestet werden kann.
UID	Eindeutiger numerischer Bezeichner zur Identifikation des Geräts in der Schnittstelle PLC ↔ HMI.
VERSION	Bezeichner für eine Version

Funktionsausführung negativ quittieren

Das System bietet mit der automatisch bereitgestellten Variablen "$actionresult" die Möglichkeit, dem XML-Parser ein negatives Ausführungsergebnis mitzuteilen. Wird der Wert auf Null gesetzt, bricht der Parser die Funktionsbearbeitung ab.

Beispiel

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE AGM>
<AGM>                                   Bezeichner für Easy Extend
<DEVICE>
 <NAME> Device 1 </NAME>                Bezeichner für das Gerät
 <START_UP>                             Bezeichner zur Inbetriebnahme des Geräts
 …
 </START_UP>
```

```
 <SET_ACTIVE>                        Bezeichner zum Aktivieren des Geräts
 …
 </SET_ACTIVE>
 <SET_INACTIVE>                      Bezeichner zum Deaktivieren des Geräts
 …
 </SET_INACTIVE>
 <TEST>                              Bezeichner zum Testen des Geräts
 …
 </TEST>
</DEVICE>
…
</AGM>
```

9.7.3 CONTROL_RESET

Beschreibung

Dieser Bezeichner erlaubt es, eine oder mehrere Steuerungskomponenten neu zu starten. Die Abarbeitung des Skripts wird erst fortgesetzt, wenn die Steuerung den zyklischen Betrieb wieder aufgenommen hat.

Programmierung

Bezeichner:	CONTROL_RESET	
Syntax:	<CONTROL_RESET resetnc="TRUE" />	
Attribute:	resetnc="true"	Die NC-Komponente wird neu gestartet.
	resetdrive="true"	Die Antriebskomponenten werden neu gestartet.

9.7.4 DATA

Beschreibung

Bezeichner für den Zugriff auf NCK, PLC und Antriebsdaten.

Weitere Einzelheiten sind beschrieben im Kapitel: Adressierung der Parameter (Seite 284)

Programmierung

Bezeichner:	DATA	
Syntax:	<data name ="<Adresse>" > Wert </data>	
Attribute:	name	Bezeichner für den Variablennamen

9.7.5 DATA_ACCESS

Beschreibung

Der Bezeichner steuert das Verhalten des Dialogs beim Speichern von Benutzereingaben. Das Verhalten ist innerhalb des Bezeichners INIT festzulegen. Wird dieser Bezeichner nicht verwendet, erfolgt immer das Zwischenspeichern der Eingaben.

Ausnahme: Folgendes Attribut ist gesetzt: hotlink = true

Programmierung

Bezeichner:	DATA_ACCESS	
Syntax:	--	
Attribute:	type=true	Es erfolgt kein Zwischenspeichern der Eingabewerte. Der Dialog kopiert die eingegebenen Werte direkt in die Referenzvariablen.
	type=false	Die Werte werden erst mit dem Bezeichner UPDATA_DATA type = "FALSE" in die Referenzvariable kopiert.

9.7.6 DATA_LIST

Beschreibung

Dieser Bezeichner erlaubt das temporäre Sichern oder Wiederherstellen der Antriebs- und Maschinendaten. Pro Gerät können bis zu 20 temporäre Datenlisten angelegt werden.

Hinweis

Mit dem Verlassen der Funktion Easy Extend löscht das System die Datenlisten.

Programmierung

Bezeichner:	DATA_LIST	
Syntax:	<DATA_LIST action ="<read/write>" id="<Listenbezeichner>" > </DATA_LIST>	
Attribute:	action	Bezeichner für Variablenwerte:
	• action="read"	• Die Werte der aufgelisteten Variablen werden in einem temporären Speicher abgelegt.
	• action="append"	• Die Werte der aufgelisteten Variablen werden an eine bestehende Liste angefügt.
	• action="write"	• Die gesicherten Werte der Variablen werden in die entsprechenden Maschinendaten kopiert.
	id	Bezeichner für die Identifikation des temporären Speichers

Beispiel

```
<DATA_LIST action ="read" id="<name>">
nck/channel/parameter/r[2]
nck/channel/parameter/r[3]
nck/channel/parameter/r[4]
$MN_USER_DATA_INT[0]
…
</DATA_LIST >
<DATA_LIST action ="write" id="<name>" />
```

9.7.7 DRIVE_VERSION

Beschreibung

Bezeichner für die Antriebsversion. Die Versionsnummer wird in die Variable `$driveversion` kopiert, die innerhalb des Bezeichners "DEVICE" gültig ist.

Weitere Einzelheiten sind beschrieben im Kapitel: Adressierung der Parameter (Seite 284)

Programmierung

Bezeichner:	DRIVE_VERSION
Syntax:	--
Attribute:	--

9.7.8 FILE

Beschreibung

Der Bezeichner ermöglicht das Einlesen oder Erstellen von Standard- oder Datenklassenarchiven.

- Archiv einlesen:

  Zum Einlesen eines Archivs ist der Dateiname des Archivs anzugeben.
- Archiv erstellen:

  Ist das Attribut create= "true" angegeben, erstellt die Funktion ein Standardarchiv (*.arc) unter dem angegebenen Namen und legt die Datei im Verzeichnis …/dvm/archives ab.

  Wird zusätzlich das Attribut class verwendet, erstellt das System ein Datenklassenarchiv. Den Inhalt legen die Attribute class und group fest.

Programmierung

Bezeichner:	FILE	
Syntax:	<file name ="<Archivname>" />	
	<file name ="<Archivname>" create="true" class="<Datenklassen>" group="<Bereich>" />	
Attribute:	name	Bezeichner für den Dateinamen
	class	Spezifiziert die Datenklassen, die im Archiv enthalten sein sollen. Sollen mehrere Datenklassen gesichert werden, sind die Klassen durch ein Leerzeichen getrennt anzugeben. Folgende Datenklassen können angegeben werden: • user • manufacturer • individual

create	Es wird ein Inbetriebnahmearchiv unter dem angegebenen Namen im Verzeichnis `…/dvm/archives/` angelegt. Ist das Attribut "class" nicht angegeben, handelt es sich um ein Standard-Archiv mit NC- /PLC- HMI- und Antriebsdaten.
group	Spezifiziert die Datengruppen, die im Archiv enthalten sein sollen. Sollen mehrere Datengruppen gesichert werden, sind die Gruppen durch ein Leerzeichen getrennt anzugeben. Folgende Datengruppen können im Archiv enthalten sein: • NC • PLC • HMI • DRIVES

Beispiel

```
<!-- Datenklassenarchiv erstellen -->
<file name="user.arc" create="true"
class="user manufacturer individual"
group="nc plc hmi" />

<!—Archiv in die Steuerung einlesen à
<file name="user.arc" />
; oder
<file name="user.ard" />
```

9.7.9 FUNCTION

Beschreibung

Funktionsaufruf: Dieser Bezeichner führt die unter dem Attribut "name" angegebene Funktion aus.

Programmierung

Bezeichner:	**FUNCTION**	
Syntax:	<FUNCTION name = "function name" />	
Attribute:	name	Name der Funktion
	return	Variablenname zum Speichern des Funktionsergebnisses
Aufrufparameter:	Aufrufparameter werden als Werte der XML-Anweisung übergeben. Die aufgeführten Variablen sind durch ein Komma voneinander zu trennen. Es können maximal 10 Parameter übergeben werden. Weiterhin ist es möglich, Konstanten oder Textausdrücke als Aufrufparameter anzugeben. Zum Kennzeichnen eines Textausdruckes ist dem Text die Kennung _T voranzustellen.	

Beispiel

Die aufrufende Funktion erwartet keinen Rückgebewert.

```
<FUNCTION name = "function name" return="variable name" />
; Parameterübergabe
<FUNCTION name = "function name"> var1, var2, var3 </FUNCTION>
<FUNCTION name = "function name"> _T"Text", 1.0, 1 </FUNCTION>
```

9.7.10 FUNCTION_BODY

Beschreibung

Funktionskörper: Dieser Bezeichner beinhaltet bildet den Funktionskörper einer Unterfunktion.

Programmierung

Bezeichner: **FUNCTION_BODY**

Syntax:

- Funktionskörper ohne Parameter

```
<FUNCTION_BODY name = "function name" >
…
</ FUNCTION_BODY>
```

- Funktionskörper mit Parameter

```
<FUNCTION_BODY name="function_name" parameter="p1, p2, p3"
>
…
<let name="tmp></let>
<op> tmp = p1 </op>
…
</FUNCTION_BODY>
```

- Funktionskörper mit Rückgabewert

```
<FUNCTION_BODY name="function_name" parameter="p1, p2, p3"
return="true" >
…
<let name="tmp></let>
<op> tmp = p1 </op>
…
<op> $return= tmp </op>
</FUNCTION_BODY>
```

Attribute:

name	Name des Funktionskörpers der Unterfunktion
return	Ist das Attribut auf true gesetzt, legt das System die lokale Variable $return an. In diese ist der Rückgabewert der Funktion zu kopieren, der mit dem Verlassen des Funktionskörpers an die aufrufende Funktion weitergeleitet wird.
(optional)parameter	Das Attribut listet die erwarteten Übergabeparameter auf. Die Parameter sind durch ein Komma voneinander zu trennen. Beim Aufruf des Funktionskörpers werden die Werte der im Funktionsaufruf angegebenen Parameter in die aufgeführten Übergabeparameter kopiert.

Beispiel

```
<function_body name="test" parameter="c1,c2,c3" return="true">
       <let name="tmp">0</let>
       <op> tmp = c1+c2+c3 </op>
       <op> $return = tmp </op>
</function_body>
…
       <let name="my_var"> 4 </let>
<function name="test" return=" my_var " > 2, 3, 4</function>
       <print text="result=%d"> my_var </print>
       …
       <op> tmp = c1+c2+c3 </op>
       <op> $return = tmp </op>
</function_body>
       <let name="my_var"> 4 </let>
<function name="test" return=" my_var " > 2, 3, 4</function>
       <print text="result=%d"> my_var </print>
       …
```

9.7.11 INCLUDE

Beschreibung

In dieser Anweisung wird eine XML-Beschreibung eingeschlossen.

Programmierung

Bezeichner:	INCLUDE	
Syntax:	<?include src="Dateiname" ?>	
Attribute:	src	Bezeichner für den Dateinamen

9.7.12 LET

Beschreibung

Bezeichner für eine lokale Variable unter dem angegebenen Namen.

Die Gültigkeit der Variable erstreckt sich bis zum Ende des übergeordneten XML-Blocks.

Variablen, die global zur Verfügung stehen sollen, sind direkt nach dem Tag AGM anzulegen.

Hinweis

Vorbelegung einer String-Variablen

Einer String-Variablen lassen sich mehrzeilige Texte zuweisen, indem man den formatierten Text als Wert übergibt. Soll ein Zeile mit einem Line Feed <LF> abgeschlossen werden, sind die Zeichen ‚\n' am Ende der Zeile anzufügen.

Programmierung

Bezeichner:	**LET**	
Syntax:	<let name ="<Name>" > Vorbelegung </let>	
	<let name ="<Name>" type="<Variablentyp>"> Vorbelegung </let>	
Attribute:	name	Bezeichner für den Variablennamen
	type	Zulässige Bezeichner für den Variablentyp: • Integer (INT) • Double (DOUBLE) • Float (FLOAT) • String (STRING) **Voreinstellung:** Wird kein Typ definiert, legt das System eine Integer-Variable an. <LET name = "VAR1" type = "INT" /> Eine Variable kann einem Wert initialisiert werden. <LET name = "VAR1" type = "INT" > 10 </LET> Werden Werte aus NCK- oder PLC-Variablen in einer lokalen Variable abgelegt, passt die Zuweisungsoperation das Format automatisch an das Format der eingelesenen Variablen an.
	permanent	Ist das Attribut auf TRUE gesetzt, wird der Variablenwert dauerhaft gespeichert. Dieses Attribut wirkt nur auf globale Variablen! **Syntax:** <let name ="<Name>" type="<Variablentyp>" permanent="TRUE" > Vorbelegung </let>

Beispiel

```
<LET name="text" type = "string"> F4000 G94\n
   G1 X20\n
   Z50\n
   M2\n
</LET>
```

9.7.13 MSGBOX

Beschreibung

Der Bezeichner öffnet ein Meldefenster, dessen Rückgabewert zur Verzweigung genutzt werden kann. Wird eine Textreferenz verwendet, zeigt das Meldefenster den für den Bezeichner hinterlegten Text an.

Programmierung

Bezeichner:	**MSGBOX**	
Syntax:	<MSGBOX text="<Text>" caption="<Überschrift>" retvalue="<Variable>" type="<Quittung>" /> <MSGBOX text="<Text>" caption="<$$Textreferenz>" retvalue="<Variable>" type="<Quittung>" />	
Attribute:	caption	Bezeichner für den Titel des Meldefensters
	retvalue	Bezeichner für den Namen der Variablen, in die der Rückgabewert kopiert wird:
	retval=0	0: OK
	retval=1	1: Cancel
	type	Bezeichner zum Quittieren der Meldung
	type="btn_ok"	
	type="btn_cancel"	
	type="btn_okcancel"	

9.7.14 OP

Beschreibung

Bezeichner für eine Ausführungsanweisung: Alle zulässigen Operatoren können ausgeführt werden. Für den Zugriff auf die NCK-, PLC- und Antriebsdaten ist der vollständige Variablenname in Anführungszeichen zu setzten.

Programmierung

Bezeichner:	OP
Syntax:	<op> Rechenoperation </op>
Attribute:	--

Beispiel

```
<OP> tmpVar = "PLC/MB170" </OP>
<OP> tmpVar = "PLC/MB170" + 5 </OP
```

Zeichenkettenverarbeitung

Die Operationsanweisung ist in der Lage, Zeichenketten zu verarbeiten und die resultierende Zeichenkette der in der Gleichung angegebenen String-Variablen zu zuweisen.

Zum Kennzeichnen von Textausdrücken ist dem Text die Kennung _T voranzustellen. Weiterhin ist das Formatieren von Variablenwerten möglich. Die Formatierungsvorschrift ist mit der Kennung _F einzuleiten, gefolgt von der Formatanweisung.

Anschließend wird die Adresse der Variablen angegeben.

Beispiel

```
<LET name="buffer" type="string"></LET>
…
…
<op> buffer = _T"unformatted value R0= " + "nck/Channel/Parameter/R[0]" + _T" and "
+ _T"$$85051" + _T" formatted value R1 " + _F%9.3f"nck/Channel/Parameter/R[1]" </op>
```

9.7.15 OPTION_MD

Beschreibung

Der Bezeichner erlaubt das Umdefinieren eines Options-Maschinendatums. Im Auslieferungszustand verwendet das System die MD14510 $MN_USER_DATA_INT[0] bis $MN_USER_DATA_INT[3].

Verwaltet das PLC-Anwenderprogramm die Optionen, sind die entsprechenden Datenwörter in einem Datenbaustein oder GUD bereitzustellen.

Das Datum ist bitweise organisiert. Beginnend mit Bit 0 erfolgt eine feste Zuordnung der Bits zur Auflistungsreihenfolge der Geräte, d.h. Bit 0 ist dem Gerät 1 zugeordnet, Bit 1 dem Gerät 2 usw. Sollen mehr als 16 Geräte verwaltet werden, erfolgt die Zuweisung der Adressbezeichner der Gerätegruppen 1-3 über den Bereichsindex.

Hinweis

Wertebereich umrechnen

Der Wertebereich des MD14510 $MN_USER_DATA_INT[i] ist von -32768 bis +32767. Um die Geräte bitweise über den Maschinendaten-Dialog freischalten zu können, ist eine Umrechnung der Bitkombination in die Dezimaldarstellung notwendig.

Programmierung

Bezeichner:	**OPTION_MD**	
Syntax:	Bereich 0:	
	<option_md name = "Adressbezeichner des Datums" />	
	ODER:	
	<option_md name = "Adressbezeichner des Datums" index= "0"/>	
	Bereich 1 bis 3:	
	<option_md name = "Adressbezeichner des Datums" index= "Bereichsindex"/>	
Attribute:	name	Bezeichner für die Adresse, z. B. $MN_USER_DATA_INT[0]
	index	Bezeichner für den Bereichsindex:
		0 (Voreinstellung): Gerät 1 bis 16
		1: Gerät 17 bis 32
		2: Gerät 33 bis 48
		3: Gerät 49 bis 64

9.7.16 PASSWORD

Beschreibung

Ist dieser Bezeichner einem Gerät zugeordnet, erscheint bei nicht gesetzter Option ein Softkey, der die Eingabe eines Passworts für dieses Gerät verlangt. Die Zeichenkette wird durch die PLC verarbeitet und das Ergebnis dem HMI über das Optionsdatum mitgeteilt.

Programmierung

Bezeichner:	**PASSWORD**	
Syntax:	<password refvar = "variable name" />	
Attribute:	refvar	Name der Referenzvariablen

Beispiel:

```
<password refvar="plc/db9900.dbd0" />
```

9.7.17 PLC_INTERFACE

Beschreibung

Der Bezeichner erlaubt das Umdefinieren der PLC ↔ HMI-Schnittstelle. Vom System werden 128 adressierbare Wörter erwartet.

Voreinstellung: DB9905

Programmierung

Bezeichner:	**PLC_INTERFACE**	
Syntax:	<plc_interface name = "Adressbezeichner des Datums" />	
Attribute:	name	Bezeichner für die Adresse, z. B. "plc/mb170"

Beispiel: `plc/mb170`

9.7.18 POWER_OFF

Beschreibung

Bezeichner für eine Meldung, die den Bediener zum Ausschalten der Maschine auffordert. Der Meldetext ist fest im System hinterlegt.

Programmierung

Bezeichner: POWER_OFF
Syntax: <power_off />
Attribute: --

9.7.19 PRINT

Beschreibung

Der Bezeichner gibt einen Text in der Meldezeile aus oder kopiert diesen in die angegebene Variable. Enthält der Text Formatierungskennungen, werden die Werte der Variablen an der entsprechenden Stelle eingefügt.

- Die Zeichenfolge "%n" führt zu einem Zeilenumbruch im angezeigten Text.
- Das ‚%' – Zeichen führt zur Formatierung der als Wert angegebenen Variable:

```
%[Flags] [Breite] [.Nachkommastellen] Typ
```

Pararmeter	Anwendung
Flags	Optionales Zeichen zum Festlegen der Formatierung für die Ausgabe: • rechts- oder linksbündig (- linksbündig) • Vornullen hinzufügen (0) • mit Leerzeichen auffüllen (blank)
Breite	Das Argument legt die minimale Ausgabebreite einer nicht negativen Zahl fest. Besitzt der auszugebende Wert weniger Stellen als durch das Argument festgelegt, werden die fehlenden Stellen mit Leerzeichen aufgefüllt. Nachkommastellen: Bei einer Gleitkommazahl legt der optionale Parameter die Anzahl der Nachkommastellen fest.

Pararmeter	Anwendung
Typ	Das Typzeichen legt fest, welche Datenformate der Anweisung PRINT übergeben wurden. Dieses Zeichen muss angegeben werden. Folgende Datenformate werden unterstützt: • d: Integerwert • f: Gleitkommazahl • s: String
Werte	Anzahl der Variablen, deren Werte in den Text eingefügt werden sollen. Die Variablentypen müssen mit der entsprechenden Typkennzeichnung der Formatierungsvorschrift übereinstimmen.

Programmierung

Bezeichner: **PRINT**

Syntax: <print name = "Variablenname" text="text %Formatierung"> Variable, … </print>

Attribute:

name	Name der Variablen, in die der Text abgelegt werden soll.
text	Text

9.7.20 WAITING

Beschreibung

Nach einem Reset der NC oder des Antriebs wird auf den Wiederanlauf der jeweiligen Komponente gewartet.

Programmierung

Bezeichner: **WAITING**

Syntax: <WAITING WAITINGFORNC ="TRUE" />

Attribute:

waitingfornc="true"	Es wird auf den Wiederanlauf der NC gewartet.
waitingfordrive="true"	Es wird auf den Wiederanlauf des Antriebs gewartet.

9.7.21 ?up

Beschreibung

SinuCom Installer:

Der Abschnitt enthält die Skript-Sprache des SinuCom Installer. Soll der Code einer Datei des SinuCom Installer inkludiert werden, ist die Anweisung INCLUDE (Seite 267) zu verwenden.

Programmierung

Bezeichner:	**?up**
Syntax:	<?up <?include src="Dateiname" ?> ?>
Attribute:	--

9.7.22 XML Bezeichner für den Dialog

Dialog zur Parametrierung

Um zusätzliche Parameter zur Laufzeit zu setzen oder auszugeben, kann für jedes Gerät ein Dialog projektiert werden. Dieser wird mit dem Drücken des Softkeys "Zusätzliche Parameter" angezeigt.

Folgende Dialogelemente sind verfügbar:

- Eingabedialog
- Dialogtitel
- Kombiniertes Ein-/Ausgabefeld
- Textanzeige
- Bildanzeige

Beschreibung

Bezeichner <tag>	Bedeutung
CAPTION	Bezeichner für den Titel des Dialogs: Syntax: <caption> Überschrift </caption>
CLOSE	**Dialognachricht:** Vor dem Schließen des Dialogs wird dieser Bezeichner abgearbeitet.

Bezeichner <tag>	Bedeutung
FORM	Bezeichner für einen Anwender-Dialog. Attribut color: Farb-Codierung der Hintergrundfarbe
INIT	Dialognachricht: Bezeichner zur Initialisierung des Dialogs. Der Bezeichner wird unmittelbar nach dem Erstellen des Dialogs abgearbeitet. Hier sind alle Eingabeelemente sowie Hotlinks des Dialogs anzulegen.
PAINT	Dialognachricht: Bezeichner zur Anzeige aller Texte und Bilder des Dialogs. Dieser Bezeichner wird beim Aufblenden des Dialogs abgearbeitet.
TIMER	Dialognachricht: Dieser Bezeichner wird zyklisch aufgerufen.

Beispiel

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE AGM>
<AGM>
<DEVICE>
 <NAME> Device 1 </NAME>
 <START_UP>
 …
 </START_UP>
 <SET_ACTIVE>
 …
 </SET_ACTIVE>
…
 <FORM>                                  Bezeichner für einen Anwender-Dialog
 <INIT>
 <CONTROL name = edit1 /CONTROL>         Bezeichner für ein Eingabefeld
 </INIT>
 <PAINT>                                 Bezeichner für Text- oder Bildanzeige
 <TEXT>hello world !</TEXT>
 </PAINT>
 </FORM>

</DEVICE>
…
</AGM>
```

9.7.23 BOX

Beschreibung

Bezeichner zum Zeichnen eines gefüllten Rechtecks an der angegebenen Position in der angegebenen Farbe.

Programmierung

Bezeichner:	BOX	
Syntax:	<box xpos="X- Position" ypos = "Y - Position" width="X-Ausdehnung" height="Y-Ausdehnung" color="Farb-Code" />	
Attribute:	xpos	Position in X-Richtung (Nullpunkt in der linken oberen Ecke)
	ypos	Position in Y-Richtung (Nullpunkt in der linken oberen Ecke)
	width	Ausdehnung in X-Richtung (in Pixel)
	height	Ausdehnung in Y-Richtung (in Pixel)
	color	Farbe

9.7.24 CONTROL

Beschreibung

Bezeichner zum Erstellen von Steuerelementen.

Voreinstellung: fieldtype="edit" Die Daten sind editierbar.

Programmierung

Bezeichner:	CONTROL	
Syntax:	<control name = "edit1" xpos = "X-Position" ypos = "Y-Position" refvar="NC-Variable" hotlink="true" format="Format" display_format="FLOAT" />	
Attribute:	name	Name des Feldes: Für jedes Feld wird eine lokale Variable gleichen Namens angelegt.
	xpos	Position in X-Richtung (Nullpunkt in der linken oberen Ecke)
	ypos	Position in Y-Richtung (Nullpunkt in der linken oberen Ecke)

fieldtype • fieldtype="edit" • fieldtype="readonly" • fieldtype="combobox"	Feldtyp: • Die Daten sind editierbar. • Die Daten sind lesbar. • Anstelle der Daten werden Bezeichner angezeigt.
format	Das Attribut definiert das Anzeigeformat der angegebenen Variablen, z. B. linksbündig oder Anzahl der Nachkommastellen.
display_format	Das Attribut definiert das Verarbeitungsformat der angegebenen Variablen. Dieses Attribut muss bei einem Zugriff auf eine PLC-Float Variable angewendet werden, da der Zugriff über das Lesen eines Doppelwortes erfolgt. Folgende Datenformate sind zulässig: • FLOAT • INT • DOUBLE • STRING
hotlink	Bezeichner für ein Feld, das bei einer Änderung der Daten sofort aktualisiert (=TRUE) wird.
refvar	Bezeichner für eine Referenzvariable (NC-, PLC- oder Antriebs-Variable)

Hinweis

Zyklische Aktualisierung

Das Attribut "hotlink" führt zu einer zyklischen Aktualisierung des entsprechenden Controls.

Das bedeutet: Wird ein Wert eingegeben, überschreibt der nachfolgende Aktualisierungszyklus den eingegebenen Wert.

Um diesen Verhalten zu umgehen, ist mit dem Bezeichner DATA_ACCESS das sofortige Speichern der Eingaben zu aktivieren.

Beispiel

Wird als Feldtyp "combobox" (= Kombinationslistenfeld) ausgewählt, sind zusätzlich die darzustellenden Ausdrücke festzulegen. Dafür ist der Bezeichner <item> zu verwenden:

Das Kombinationslistenfeld speichert den Index des aktuell ausgewählten Textes in der zum "CONTROL" gehörenden Variablen (Variablenname). Der Index beginnt mit dem Wert 1.

Syntax: <item>Ausdruck</item>

```
<control name = "button1" xpos = "10" ypos = "10" fieldtype="combobox">
      <item>text1</item>
      <item>text2</item>
      <item>text3</item>
      <item>text4</item>
</control>

      ; Soll ein beliebiger Integer - Wert einem Ausdruck zugeordnet werden, ist
      das Attribut value="Wert" dem Bezeichner item hinzuzufügen.
      ; Anstelle der fortlaufenden Nummerierung enthält jetzt die Control-Variable
      den zugewiesenen Wert des Items.

<control name = "button1" xpos = "10" ypos = "10" fieldtype="combobox">
      <item value = "10" >text1</item>
      <item value = "20" >text2</item>
      <item value = "12" >text3</item>
      <item value = "1" >text4</item>
</control>
```

9.7.25 IMG

Beschreibung

Bezeichner zum Anzeigen einer Pixel-Grafik im Verzeichnis: `../oem/sinumerik/hmi/dvm`

- Die Bitmaps sind im BMP- oder PNG-Format abzulegen.
- Für alle Dateinamen sind Kleinbuchstaben zu verwenden.
- Soll die Darstellung des Bildes von der Originalgröße abweichen, kann die Dimension mit den Attributen width und height festgelegt werden.

Programmierung

Bezeichner: **IMG**

Syntax: <img name = "<Name>" xpos = " X- Position " ypos = " Y - Position " height=" Skalierung in Y" width="Skalierung in X" />

Attribute:	name	Ausdehnung in Y-Richtung (in Pixel)
	xpos	Position in X-Richtung (Nullpunkt in der linken oberen Ecke)
	ypos	Position in Y-Richtung (Nullpunkt in der linken oberen Ecke)
	width	Skalierung in X (optional)
	height	Skalierung in Y (optional)

9.7.26 PROPERTY

Beschreibung

Der Bezeichner legt zusätzliche Eigenschaften eines Bedienelementes fest.

Programmierung

Bezeichner:	**PROPERTY**	
Syntax:	<property Attribut="<Wert>" />	
Attribute:	max	Maximaler Eingabewert
	min	Minimaler Eingabewert
	default	Vorbelegung

Beispiel

```
<control name = "edit" xpos = "10" ypos = "10" width = "100" hotlink="true"
refvar="nck/Channel/GeometricAxis/actProgPos[1]" >
       <property min="0" />
       <property max="1000" />
</control>

<control name = "edit1" xpos = "10" ypos = "10" >
       <property min = "20" />
       <property max = "40" />
       <property default="25" />
</control>
```

9.7.27 REQUEST

Beschreibung

Dieser Bezeichner ist nur innerhalb der INIT-Anweisung einer Form gültig. Mit Hilfe dieses Bezeichners wird eine Variable in den zyklischen Lesedienst (hotlink) aufgenommen.

Programmierung

Bezeichner:	**REQUEST**	
Syntax:	<REQUEST name = "NC-Variable" />	
Attribute:	name	Adressbezeichner

9.7.28 SOFTKEY_OK, SOFTKEY_CANCEL

Beschreibung

Der Bezeichner SOFTKEY_OK überschreibt das Standardverhalten beim Schließen eines Dialogs mittels Softkey "OK". Der Bezeichner SOFTKEY_CANCEL überschreibt das Standardverhalten beim Schließen eines Dialogs mittels Softkey "CANCEL".

Innerhalb dieser Bezeichner können folgende Funktionen ausgeführt werden:

- Datenmanipulationen
- Bedingte Bearbeitung
- Schleifenbearbeitung

Programmierung

Bezeichner:	**SOFTKEY_OK**
Syntax:	<SOFTKEY_OK> … </SOFTKEY_OK>
Bezeichner:	**SOFTKEY_CANCEL**
Syntax:	<SOFTKEY_CANCEL> … </SOFTKEY_CANCEL>

9.7.29 TEXT

Beschreibung

Bezeichner zum Anzeigen von Text

Programmierung

Bezeichner:	TEXT	
Syntax:	<text xpos="X- Position" ypos = "Y - Position" color="Farb-Code"> Text </text>	
Attribute:	xpos	Position in X-Richtung (Nullpunkt in der linken oberen Ecke)
	ypos	Position in Y-Richtung (Nullpunkt in der linken oberen Ecke)
	color	Farbe

9.7.30 TYPE_CAST

Beschreibung

Bezeichner zum Umwandeln des Datentyps einer lokalen Variablen.

Programmierung

Bezeichner:	TYPE_CAST	
Syntax:	<type_cast name="variable name" type=" new type" />	
Attribute:	name	Name der Variablen
	type	neuer Datentyp
	convert	interpretiert die Bitkombination (indirekte Zuweisung)

Beispiel

```
<let name="v1" type="string" >1234.566</let>
<let name="result" type="double" />
…
<type_cast name="v1" type="double" />
<op> result = v1 *23.5 </op>
…
```

Sonderfall:

Soll ein PLC-Doppelwort als Gleitkommazahl verarbeitet werden, ist eine Zuweisung auf eine lokale Variable notwendig. Damit der Binärwert korrekt gespeichert werden kann, muss das Format der lokalen Variablen von Typ Integer sein.

Anschließend ist mittels der Anweisung TYPE_CAST die Konvertierung in eine Gleitkommazahl durchzuführen. Um den Integer-Wert als Gleitkommazahl interpretieren zu können, verwendet man das Attribut convert anstelle des Attributs type. Es interpretiert die Bitkombination des Integer-Wertes als Float-Wert und ändert die Formatfestlegung der Variablen zu float.

```
<let name="plc_float" />
<let name="result" type="float" />
…
…
<timer>

   <type_cast name="plc_float" type="int" />
   <op> plc_float = "PLC/MD100"</op>
   <type_cast name="plc_float" convert="float" />

   <op> result = plc_float *1.2 </op>
</timer>
```

9.7.31 UPDATE_CONTROLS

Beschreibung

Dieser Bezeichner führt den Abgleich zwischen den Bedienelementen und den Referenzvariablen durch.

Programmierung

Bezeichner:	UPDATE_CONTROLS	
Syntax:	<update_controls type="<Richtung>"/>	
Attribute:	type	Das Attribut legt die Richtung des Datenabgleichs fest. • TRUE Die Daten werden aus den Referenzvariablen gelesen und in die Bedienelemente kopiert. • FALSE Die Daten werden aus den Bedienelementen in die Referenzvariablen kopiert.

9.7.32 Adressierung der Parameter

Parameter adressieren

Zum Adressieren von NC-Variablen, PLC-Bauseinen oder Antriebsdaten sind Adressbezeichner auf das gewünschte Datum zu bilden. Eine Adresse besteht aus den Teilpfaden Komponentenname und Variablenadresse. Als Trennzeichen ist ein Schrägstrich zu verwenden.

Adressierung von PLC-Datenbausteinen:

Datenformat f:	B: Byte
	W: Wort
	D: Doppelwort
Adresse x:	gültiger PLC-Adressbezeichner
Bit-Adressierung:	b = Bitnummer

Folgende Adressen sind zulässig:

DBx.DB(f)	Datenbaustein
`I(f)x`	Eingang
`Q(f)x`	Ausgang
`M(f)x`	Merker
`V(f)x`	Variable

Bei einer Bit-Adressierung entfällt die Datenformatkennzeichnung:

DBx.DBXx.b	Datenbaustein
`Ix.b`	Eingang
`Qx.b`	Ausgang
`Mx.b`	Merker
`Vx.b`	Variable

Beispiel:

```
<data name = "plc/mb170">1</data>

<data name = "plc/db9905.dbb0"> 0 </data>

<data name = "plc/i0.1"> 1 </data>

<op> plc/m19.2 = 1 </op>
```

Adressierung von NC-Variablen

Die Adressierung beginnt mit dem Pfadanteil nck. Diesem Anteil folgt die Adresse des Datums, deren Aufbau dem OEM-Paket Listenbuch 2 zu entnehmen ist.

Beispiel:

```
<let name = "tempStatus"></let>

<op> tempStatus = "nck/channel/state/chanstatus" </op>
```

Adressierung der Maschinendaten und Setting-Daten

Maschinendaten und Setting-Daten werden mit dem Zeichen $ gekennzeichnet, gefolgt vom Namen des Datums.

- Maschinendaten:

  $Mx_<Name[index, AX<Achsnummer>]>
- HMI-Maschinendaten:

  $MxS_<Name[index, AX<Achsnummer>]>
- Optionsdaten

  $Ox_<Name[index, AX<Achsnummer>]>
- Setting-Daten:

  $Sx_<Name[index, AX<Achsnummer>]>

  $SxS_<Name[index, AX<Achsnummer>]>

Aufbau		Bedeutung
x:	N	allgemeine Maschinen- oder Setting-Daten
	C	kanalspezifische Maschinen- oder Setting-Daten
	A	achsspezifische Maschinen- oder Setting-Daten
index:		Bei einem Feld gibt der Parameter den Index des Datums an.
AX<Achsnummer>:		Bei achsspezifischen Daten ist die gewünschte Achse (<Achsnummer>) zu spezifizieren. Alternativ kann der Achsindex mittels "Ersetzungszeichen" $<Variablenname> aus einer lokalen Variablen gelesen werden: z. B. AX$lokaleVariable

Beispiel:

```
<data name ="$MN_AXCONF_MACHAX_NAME_TAB[0] ">X1</data>
```

- Direkte Adressierung der Achse:

  ```
  <data name ="$MA_CTRLOUT_MODULE_NR[0, AX1] ">1</data>
  ```

- Indirekte Adressierung der Achse

  ```
  <let name ="axisIndex"> 1 </let>

  <data name ="$MA_CTRLOUT_MODULE_NR[0, AX$axisIndex] ">1</data>
  ```

Adressierung der globalen Anwenderdaten

Die Adressierung beginnt mit dem Pfadanteil gud, gefolgt von der Bereichsspezifikation CHANNEL. Diesem Adressteil folgt die Spezifizierung der GUD-Bereiche:

GUD Bereiche	Zuordnung
sgud	GUD von Siemens
mgud	GUD des Maschinenherstellers
ugud	GUD des Anwenders

Anschließend ist der GUD-Name anzugeben. Ist ein Feld zu adressieren, folgt dem Namen der Feldindex in eckigen Klammern.

Beispiel:

```
<data name ="gud/channel/mgud/syg_rm[0]">1</data>

<op>"gud/channel/mgud/syg_rm[0]" = 5*2 </op>
```

9.7.33 Adressierung der Antriebsobjekte

Syntax für Drive Objekte (DO)

Die Adressierung beginnt mit dem Pfadanteil "drive". Es folgt die Spezifizierung des Antriebgerätes: CU oder DC. Diesem Anteil wird der zu setzende Parameter angefügt.

Zum Adressieren der einzelnen Objekte ist nach dem Parameter das gewünschte Objekt in eckiger Klammer anzugeben.

```
Parameternummmer[do<DO-index>]
```

Beispiel: `p0092[do1]`

Hinweis

Nummerierung

Die Nummer des Drive Objekts weicht von der im Antriebsdialog verwendeten Nummerierung ab, da die Komponenten CU, ALM sowie alle angeschlossenen Hubs in die durchgehende Nummerierung mit eingebunden sind.

Die DO-Nummer kann wie folgt ermittelt werden:

Alle angeschlossenen Drive Objekte werden im Feld p0978 der entsprechenden CU aufgelistet. Dabei entspricht der Zahlenwert eines Feldes der Steckplatznummer des Drive Objektes. Es ist der Feldindex des gewünschten Steckplatzes zu ermitteln und um eins zu inkrementieren. Dieser Wert ist der für die Adressierung notwendige DO-Index.

Sind Driveobjekte an eine NX anschlossen, ist zuerst der Index des letzten CU-Drive Objektes zu ermitteln und mit dem Index des NX-Drive Objektes zu addieren.

Alternativ kann der Drive Index mittels "Ersetzungszeichen" $<Variablenname> aus einer lokalen Variablen gelesen werden, z. B. DO$lokaleVariable

Beispiel:

```
<data name ="drive/cu/p0092">1</data>

<data name ="drive/dc/p0092[do1] ">1</data>
```

Indirekte Adressierung:

```
<let name = "driveIndex> 0 </let>

<op> driveIndex = $ctrlout_module_nr[0, AX1] </op>

<data name ="drive/dc[do$driveIndex]/p0092">1</data>
```

Adressierungeiner NX

Ein NX-Modul wird wie eine weitere CU behandelt, erfolgt die Adresseierung der Module über die CU-Spezifikation, wobei nach dem Parameter der gewünschte NX-Index um eins inkrementiert in eckigen Klammern anzugeben ist.

```
Parameternummmer[CU<CU - index>]
```

Beispiel

```
<let name="r0002_content"></let>
<let name="p107_content"></let>
<!- Lesen des Wertes r0002 aus der CU ->
<op> r0002_content = "drive/cu/r0002" </op>
<op> r0002_content = "drive/cu/r0002[CU1]" </op>
<!- Lesen des Wertes r0002 aus der NX1 ->
<op> r0002_content = "drive/cu/r0002[CU2]" </op>
<!- Lesen des Wertes p107[0] aus der CU ->
<op> p107_content = "drive/cu/p107[0]" </op>
   <print text="%d"> p107_content </print>
<!- Lesen des Wertes p107[0] aus der NX1 ->
<op> p107_content = "drive/cu/p107[0,CU2]" </op>
   <print text="%d"> p107_content </print>
```

9.7.34 XML Bezeichner für Anweisungen

Programmanweisungen

Folgende Bezeichner sind für Anweisungen zulässig:

Bezeichner <tag>	Bedeutung
IF	Bedingte Anweisung (IF, THEN, ELSE) Die Tags THEN und ELSE sind in das IF-Tag eingeschlossen. Dem IF-Tag folgt die Bedingung, die im CONDITION-Tag ausgeführt wird. Das Operationsergebnis entscheidet über die weitere Bearbeitung der Anweisungen. Ist das Funktionsergebnis wahr, wird der THEN-Zweig ausgeführt und der ELSE-Zweig übersprungen. Ist das Funktionsergebnis unwahr, arbeitet der Parser den ELSE-Zweig ab. **Beispiel:** <IF> <CONDITION> plc/mb170 != 7 </CONDITION> <THEN> <OP> plc/mb170 = 7 </OP> … </TEHN> <ELSE> … … </ELSE> </IF>
THEN	Anweisung, wenn die Bedingung erfüllt wurde (IF, THEN, ELSE).
ELSE	Anweisung, wenn die Bedingung nicht erfüllt wurde (IF, THEN, ELSE).

Bezeichner <tag>	Bedeutung
FOR	Die FOR-Schleife wird wie folgt ausgeführt: 1. Der Ausdruck **Initialisierung** (INIT) wird ausgewertet. 2. Der Ausdruck **Test** (CONDITION) wird als boolescher Ausdruck ausgewertet. Falls der Wert falsch (FALSE) ist, wird die FOR-Schleife beendet. 3. Die nachfolgenden Anweisungen werden ausgeführt. 4. Der Ausdruck **Fortsetzung** (INCREMENT) wird ausgewertet. 5. Es geht mit Schritt 2 weiter. `for (Initialisierung, Test, Fortsetzung) Anweisungen` **Syntax:** <FOR> <INIT>…</INIT> <CONDITION>…</CONDITION> <INCREMENT>…</INCREMENT> Anweisungen … </FOR>
BREAK	Bedingter Abbruch einer Schleife
WHILE	Die WHILE-Schleife dient dazu, eine Abfolge von Anweisungen mehrfach auszuführen, solange eine Bedingung erfüllt ist. Diese Bedingung wird geprüft, bevor die Anweisungsfolge abgearbeitet wird. `while (Test) Anweisungen` **Syntax:** <WHILE> <CONDITION>…</CONDITION> Anweisungen … </WHILE>

DO_WHILE	Die DO ... WHILE-Schleife besteht aus einem Anweisungsblock und einer Bedingung. Der Code innerhalb des Anweisungsblocks wird zuerst ausgeführt, anschließend die Bedingung ausgewertet. Falls die Bedingung wahr (TRUE) ist, führt die Funktion den Code-Anteil erneut aus. Dies wird solange wiederholt, bis die Bedingung falsch (FALSE) eintritt.
	
	`do`
	`     Anweisungen`
	`while (Test)`
	Syntax:
	<DO_WHILE>
	Anweisungen
	…
	<CONDITION>…</CONDITION>
	</DO_WHILE>

Bezeichner <tag>	Bedeutung
SWITCH	Die Anweisung SWITCH beschreibt eine Mehrfachauswahl. Ein Ausdruck wird einmal ausgewertet und mit einer Anzahl Konstanten verglichen. Stimmt der Ausdruck mit der Konstanten überein, werden die Anweisungen innerhalb der Anweisung CASE abgearbeitet. Die Anweisung DEFAULT wird abgearbeitet, wenn keine der aufgeführten Konstanten mit dem Ausdruck übereinstimmt. **Syntax:** <SWITCH> <condition> Ausdruck </condition> <CASE value="<Konstante 1>" > Anweisungen … </CASE> <CASE value="<Konstante 2>" > Anweisungen … </CASE> <DEFAULT> Anweisungen … </DEFAULT> </SWITCH>

9.8 String-Funktionen

Übersicht über die Funktionen

Die Skript-Sprache bietet verschiedene String-Funktionen an. Die Funktionsnamen sind reserviert und können nicht überladen werden:

Name	Funktion
string.cmp	Zeichenkette vergleichen (string.cmp (Seite 292))
string.icmp	Zeichenkette ohne Beachtung der Groß-/Kleinschreibung vergleichen (string.icmp (Seite 293))
string.left	Anzahl Zeichen von links auswählen (string.left (Seite 294))
string.right	Anzahl Zeichen von rechts auswählen (string.right (Seite 295))
string.middle	Anzahl Zeichen aus der Mitte auswählen (string.middle (Seite 295))
string.length	Länge einer Zeichenkette ermitteln (string.length (Seite 296))
string.replace	Zeichenkette ersetzen (string.replace (Seite 297))
string.remove	Zeichenkette löschen (string.remove (Seite 297))
string.insert	Zeichenkette ab Index einfügen (string.insert (Seite 299))
string.delete	Anzahl Zeichen in einer Zeichenkette löschen (string.delete (Seite 298))
string.find	Untermenge einer Zeichenkette finden (vorwärts) (string.find (Seite 299))
string.reversefind	Untermenge einer Zeichenkette rückwärts finden (string.reversefind (Seite 300))
string.trimleft	Leerzeichen von links entfernen (string.trimleft (Seite 301))
string.trimright	Leerzeichen von rechts entfernen (string.trimright (Seite 301))

9.8.1 string.cmp

Beschreibung

Es werden zwei Strings miteinander verglichen.

Die Funktion liefert den Rückgabewert Null, wenn die Strings gleich sind, kleiner Null, wenn der erste String kleiner als der zweite String ist oder größer Null, wenn der zweite String kleiner als der erste String ist.

Programmierung

Bezeichnung:	**string.cmp**	
Syntax:	<function name="string.cmp" retvar ="<int var>" > str1, str2 </function>	
Parameter:	str1	String
	str2	Vergleichsstring
	rval	Ergebnis

Beispiel

```
<let name="rval">0</let>
<let name="str1" type="string">A brown bear hunts a brown dog.</let>
<let name="str2" type="string">A brown bear hunts a brown dog.</let>
<function name="string.cmp" return="rval"> str1, str2 </function>

; Ergebnis: rval=0
```

9.8.2 string.icmp

Beschreibung

Es werden zwei Strings ohne Unterscheidung der Groß-/Kleinschreibung miteinander verglichen.

Die Funktion liefert den Rückgabewert Null, wenn die Strings gleich sind, kleiner Null, wenn der erste String kleiner als der zweite String ist oder größer Null, wenn der zweite String kleiner als der erste String ist.

Programmierung

Bezeichnung:	**string.icmp**	
Syntax:	<function name="string.icmp" retvar ="<int var>" > str1, str2 </function>	
Parameter:	str1	String
	str2	Vergleichsstring
	rval	Ergebnis

Beispiel

```
<let name="rval">0</let>
<let name="str1" type="string">A brown bear hunts a brown dog.</let>
<let name="str2" type="string">A brown Bear hunts a brown Dog.</let>
<function name="string.icmp" return="rval"> str1, str2 </function>

; Ergebnis: rval=0
```

9.8.3 string.left

Beschreibung

Die Funktion extrahiert die ersten nCount Zeichen aus String 1 und kopiert diese in die Return-Variable.

Programmierung

Bezeichnung:	**string.left**	
Syntax:	<function name="string.left" return="< result string>"> str1, nCount </function>	
Parameter:	str1	String
	nCount	Anzahl Zeichen

Beispiel

```
<let name="str1" type="string">A brown bear hunts a brown dog.</let>
<let name="str2" type="string"></let>
<function name="string.left" return="str2"> str1, 12 </function>

; Ergebnis: str2="A brown bear"
```

9.8.4 string.right

Beschreibung

Die Funktion extrahiert die letzten nCount Zeichen aus String 1 und kopiert diese in die Return-Variable.

Programmierung

Bezeichnung:	**string.right**	
Syntax:	<function name="string.right" return="< result string>"> str1, nCount </function>	
Parameter:	str1	String
	nCount	Anzahl Zeichen

Beispiel

```
<let name="str1" type="string">A brown bear hunts a brown dog.</let>
<let name="str2" type="string"></let>
<function name="string.right" return="str2"> str1, 10 </function>

; Ergebnis: str2="brown dog"
```

9.8.5 string.middle

Beschreibung

Die Funktion extrahiert ab dem Index iFirst die angegebene Anzahl Zeichen aus String 1 und kopiert diese in die Return-Variable.

Programmierung

Bezeichnung:	**string.middle**
Syntax:	<function name="string.middle" return="< result string>"> str1, iFirst, nCount </function>

Parameter:	str1	String
	iFirst	Start Index
	nCount	Anzahl Zeichen

Beispiel

```
<let name="str1" type="string">A brown bear hunts a brown dog.</let>
<let name="str2" type="string"></let>
<function name="string.middle " return="str2"> str1, 2, 5 </function>

; Ergebnis: str2="brown"
```

9.8.6 string.length

Beschreibung

Die Funktion liefert die Anzahl Zeichen eines Strings.

Programmierung

Bezeichnung:	**string.length**	
Syntax:	<function name="string.length" return="< int var>"> str1 </function>	
Parameter:	str1	String
	length	Ergebnis

Beispiel

```
<let name="length">0</let>
<let name="str1" type="string">A brown bear hunts a brown dog.</let>
<function name="string.length" return="length"> str1 </function>

; Ergebnis: length=31
```

9.8.7 string.replace

Beschreibung

Die Funktion ersetzt alle gefundenen Teil-Strings mit dem neuen String.

Programmierung

Bezeichnung:	**string.replace**	
Syntax:	<function name="string.replace"> string, find string, new string </function>	
Parameter:	string	String
	find string	zu ersetzender String
	new string	neuer String

Beispiel

```
<let name="str1" type="string">A brown bear hunts a brown dog. </let>
<function name="string.replace" > str1, _T"a brown dog" ,
_T"a big salmon"</function>

;Ergebnis: str1="A brown bear hunts a big salmon."
```

9.8.8 string.remove

Beschreibung

Die Funktion löscht alle gefundenen Teil-Strings.

Programmierung

Bezeichnung:	**string.remove**	
Syntax:	<function name="string.remove" > string, remove string </function>	
Parameter:	string	String
	remove string	zu löschender Teil-String

Beispiel

```
<let name="index">0</let>
<let name="str1" type="string">A brown bear hunts a brown dog. </let>
<function name="string.remove" > str1, _T"a brown dog" </function>

; Ergebnis: str1="A brown bear hunts ."
```

9.8.9 string.delete

Beschreibung

Die Funktion löscht ab der angegebenen Startposition die festgelegte Anzahl Zeichen.

Programmierung

Bezeichnung:	**string.delete**	
Syntax:	function name="string.delete"> string, start index , nCount </function>	
Parameter:	string	String
	start index	Startindex
	nCount	Anzahl Zeichen

Beispiel

```
<let name="str1" type="string">A brown bear hunts. </let>
<function name="string.delete" > str1, 2, 5 </function>

; Ergebnis: str1="A bear hunts."
```

9.8.10 string.insert

Beschreibung

Die Funktion fügt einen String am angegebenen Index ein.

Programmierung

Bezeichnung:	**string.insert**	
Syntax:	<function name="string.insert"> string, index, insert string </function>	
Parameter:	string	String-Variable
	index	Anzahl der Zeichen, die eingefügt werden sollen
	insert string	String, der eingefügt werden soll

Beispiel

```
<let name="str1" type="string">A brown bear hunts. </let>
<let name="str2" type="string">a brown dog </let>
<function name="string.insert"> str1, 19, str2 </function>

; Ergebnis: str1="A brown bear hunts a brown dog."
```

9.8.11 string.find

Beschreibung

Die Funktion sucht im übergebenen String nach der ersten Übereinstimmung mit dem Teil-String. Wurde der Teil-String gefunden, liefert die Funktion den Index auf das erste Zeichen (mit Null beginnend), ansonsten -1.

Programmierung

Bezeichnung:	**string.find**	
Syntax:	<function name="string.find" return="<int val>"> str1, find string </function>	
Parameter:	string	String-Variable
	find string	String, nach dem gesucht wird.

Beispiel

```
<let name="index">0</let>
<let name="str1" type="string">A brown bear hunts a brown dog. </let>
<function name="string.find" return="index"> str1, _T"brown" </function>

; Ergebnis: index=2
```

9.8.12 string.reversefind

Beschreibung

Die Funktion sucht im übergebenen String nach der letzten Übereinstimmung mit dem Teil-String. Wurde der Teil-String gefunden, liefert die Funktion den Index auf das erste Zeichen (mit Null beginnend), ansonsten -1.

Programmierung

Bezeichnung:	**string.reversefind**	
Syntax:	<function name="string.reversefind" return="<int val>"> str1, find string </function>	
Parameter:	string	String-Variable
	find string	String, nach dem gesucht wird.

Beispiel

```
<let name="index">0</let>
<let name="str1" type="string">A brown bear hunts a brown dog. </let>
<function name="string.reversefind" return="index"> str1, _T"brown" </function>

; Ergebnis: index=21
```

9.8.13 string.trimleft

Beschreibung

Die Funktion löscht führende Leerzeichen aus einem String.

Programmierung

Bezeichnung: **string.trimleft**
Syntax: <function name="string.trimleft" > str1 </function>
Parameter: str1 String-Varibale

Beispiel

```
<let name="str1" type="string">        test trim left</let>
<function name="string.trimleft"> str1 </function>

; Ergebnis: str1="test trim left"
```

9.8.14 string.trimright

Beschreibung

Die Funktion löscht nachfolgende Leerzeichen aus einem String.

Programmierung

Bezeichnung: **string.trimright**
Syntax: <function name="string.trimright" > str1 </function>
Parameter: str1 String-Variable

Beispiel

```
<let name="str1" type="string">  test trim right      </let>
<function name="string.trimright" > str1 </function>

; Ergebnis: str1=" test trim right"
```

9.9 Trigonometrische Funktionen

Übersicht über die Funktionen

Die Skript-Sprache bietet verschiedene trigonometrische Funktionen an. Die Funktionsnamen sind reserviert und können nicht überladen werden.

Trigonometrische Funktionen und Umkehrfunktionen:

Name	Funktion
sin	Sinus
cos	Cosinus
tan	Tangens
arcsin	Arcussinus
arccos	Arcuscosinus
arctan	Arcustangens

Beschreibung Sinus, Cosinus, Tangens

Die Funktion berechnet den Sinus, Cosinus, Tangens des übergebenen Wertes.

Programmierung

Bezeichnung: sin
Syntax: <function name="sin" return="<double val>"> double </function >
Bezeichnung: cos
Syntax: <function name="cos" return="<double val>"> double </function >
Bezeichnung: tan
Syntax: <function name="tan" return="<double val>"> double </function >
Parameter: double Winkel (0° bis 360°)

Beispiel

```
<let name= "sin_val" type="double"></let>
<function name="sin" return="sin_val"> 20.0 </function>
```

Beschreibung Arcussinus, Arcuscosinus, Arcustangens

Die Funktion berechnet den Arcussinus, Arcuscosinus, Arcustangens des übergebenen Wertes.

Programmierung arcsin, arccos

Bezeichnung:	**arcsin**		
Syntax:	<function name="arcsin" return="<double val>"> double </function >		
Bezeichnung:	**arccos**		
Syntax:	<function name="arccos" return="<double val>"> double </function >		
Parameter:	double	x im Bereich von -1 bis +1	
Wertebereich:		arcsin	y im Bereich von -π/2 bis +π/2
		arccos	y im Bereich von 0 bis π

Programmierung arctan

Bezeichnung:	**arctan**	
Syntax:	<function name="arctan" return="<double val>"> double </function >	
Parameter:	double	x beliebiger Wert
Wertebereich:		y im Bereich von -π/2 bis +π/2

Beispiel

```
<let name= "arccos_val" type="double"></let>
<function name="arccos" return="arctan_val"> 0.47 </function>
```

Werkzeugverwaltung 10

10.1 Grundlagen

Werkzeugverwaltung (WZV)

Die Funktion Werkzeugverwaltung (WZV) stellt sicher, dass an der Maschine zu jeder Zeit das richtige Werkzeug am richtigen Platz ist.

Maschine, Magazine, Beladestellen und Werkzeugzwischenspeicher (z. B. Spindeln, Greifer) bilden ein konkretes Anlagensystem, in welchem die Werkzeuge aufbewahrt und transportiert werden. Die Werkzeugverwaltung informiert den NCK stets über den aktuellen Standort der Werkzeuge und protokolliert die durch Teileprogramm, PLC oder HMI ausgelösten Werkzeugbewegungen.

Bei Inbetriebnahme der Werkzeugverwaltung wird die Architektur des konkreten Anlagensystems in der Steuerung abgebildet. Zum Beispiel werden ein oder mehrere Magazine angelegt, die auf ihren Plätzen Werkzeuge aufnehmen können. Der "Aufenthaltsort" eines Werkzeugs wird in der Steuerung in der Form von Wertepaaren (Magazinnummer und Platznummer) beschrieben.

Hinweis

Lieferumfang

Die Werkzeugverwaltung ist im Lieferumfang für alle Steuerungen (M-/T-Version) enthalten.

Die Funktion "Ersatzwerkzeuge für Werkzeugverwaltung" (Duplo-Werkzeuge) ist eine **Option**.

Siehe auch

Weitere Literatur:

- SINUMERIK 828D Listenhandbuch
- Eine umfassende Beschreibung der Werkzeug- und Magazinparameter sowie zur internen Datenstruktur finden Sie in:

  → Funktionsbeschreibung Werkzeugverwaltung der SINUMERIK 840D sl

  Der den NCK betreffenden Teil dieser Dokumentation ist, soweit im Funktionsumfang enthalten, auch für die SINUMERIK 828D gültig.

  Die Beschreibung der PLC Funktionen und die Kommunikation zwischen NC und PLC aus dieser Dokumentation gilt **nicht** für die SINUMERIK 828D.
- Einstellungen der Bedienoberfläche der Werkzeugverwaltung sind beschrieben in:

  → Inbetriebnahmehandbuch Basesoftware und Bediensoftware (IM9)

10.1.1 Struktur der Werkzeugverwaltung

Funktionsstruktur

Die Software-Komponenten der Steuerung haben bei der Werkzeugverwaltung folgende Aufgaben:

- HMI:
  - Werkzeugdatenanzeige, Ein-/Ausgabe
  - Magazindatenanzeige, Ein-/Ausgabe
  - Umsetzdialog Beladen/Entladen
- NCK:

  Die Werkzeugverwaltung verwaltet die Plätze der Magazine. Diese Plätze können frei, mit Werkzeugen bestückt oder durch übergroße Werkzeuge auf Nachbarplätzen belegt sein. Freie Plätze können mit weiteren Werkzeugen beladen werden. Über die Werkzeugverwaltung ist für den Maschinenhersteller eine optimale Verwaltung der Werkzeuge und Magazinplätze gegeben. Mit der Magazinverwaltung stehen erweiterte Funktionen wie Beladen, Entladen, Positionieren von Werkzeugen zur Verfügung. Außerdem gibt es Suchfunktionen von Werkzeugen, Magazinplätzen und Suchstrategien für Ersatzwerkzeuge.

  Bei den Überwachungsfunktionen werden bei Ablauf der aktiven Überwachung Werkzeuge gesperrt und nicht mehr verwendet. Für die weitere Bearbeitung wird - sofern vorhanden - automatisch ein nicht gesperrtes gleichartiges Werkzeug (Duplo-Werkzeug) verwendet.
- PLC:
  - Werkzeugwechsel durchführen
  - Werkzeuge im Magazin bewegen
  - Greifersteuerung
  - gegebenenfalls Magazinsteuerung
  - Sicherheitsverriegelungen
  - Bereitstellung der Struktur der Werkzeugbewegungen in Transferschritt-Tabellen
  - Quittung der Werkzeugbewegungen mit Quittungsschritt-Tabellen

PLC-Anwenderprogramm

Das Anwenderprogramm der PLC führt die Aufträge der Werkzeugverwaltung aus und quittiert alle Positionsänderungen von Werkzeugen (und Magazinen). Die Überwachung und Verhinderung von möglichen Kollisionen ist alleinige Aufgabe des PLC-Anwenderprogramms, z. B.:

- Mehrere Spindeln benutzen das gleiche Magazin.
- Die Wege von gleichzeitig anstehenden Aufträgen kreuzen sich.
- Solange sich ein großes Werkzeug im Shifter befindet, darf die Kette nicht bewegt werden.

PLC-Firmware

Funktionen der PLC-Firmware:

- Aufträge der Werkzeugverwaltung an das PLC-Anwenderprogramm übergeben
- Quittungen des PLC-Anwenderprogramm an die Werkzeugverwaltung übermitteln
- Rückmeldungen zu jeder Quittung (Quittung fehlerhaft mit Fehlernummer oder Quittung OK) an das PLC-Anwenderprogramm übergeben
- Zusätzlich: Auftragszustand registrieren

10.1.2 Komponenten der Werkzeugverwaltung

Werkzeugliste, Magazine, Magazinliste

Es können Revolver- und Kettenmagazine verwaltet werden. Andere Magazinarten werden auf diese abgebildet. Beladestellen oder Beladestationen sind als Magazinart für das Be- und Entladen zu verwenden.

Ein Magazin-Zwischenspeicher fasst alle weiteren Plätze zusammen, auf denen Werkzeuge platziert werden können (Spindel, Greifer, ...).

Hinweis

Voreinstellungen bei Werkzeugmagazinen

Die Anzahl der Magazine, die der NCK verwalten kann, ist vom System fest vorgegeben:

- PPU24x.2: Anzahl Magazine = 3
- PPU26x.2: Anzahl Magazine = 3
- PPU28x.2: Anzahl Magazine = 4

Da immer mindesten ein Zwischenspeicher und eine Beladestelle vorhanden sein müssen, können mit einer PPU24x.2/PPU26x.2 ein reales Magazin und mit einer PPU28x.2 zwei reale Magazine verwaltet werden.

Beim Hochlauf der Steuerung sind abhängig von der Technologie folgende Werkzeugmagazine voreingestellt und eingerichtet:

- Fräsen: ein Kettenmagazin mit 20 Plätzen, Doppelgreifer und ein Werkzeugträger
- Drehen: ein Revolver mit 8 Plätzen, ein Werkzeugträger

Magazine

Vom System werden für alle Plätze im Magazin Informationen zur Verfügung gestellt, die Inhalt und Zustand der Plätze beschreiben.

Die Position eines Werkzeugs wird durch einen Bezeichner für das Magazin und einen Bezeichner für den Platz beschrieben. Magazine haben dabei Bezeichner und Nummer, Magazinplätze nur eine Nummer. In einem realen Magazin (Kette, Revolver usw.) ist die Position des Werkzeugs durch die bei der Inbetriebnahme vergebene Magazinnummer und den Platz in diesem Magazin gekennzeichnet.

Beispiel:

Die T-Nummer des Werkzeugs des Magazinplatzes 7 aus Magazin 1: $TC_MPP6[1,7]

Werkzeugliste

Die Werkzeugliste enthält alle der NC bekannten Werkzeuge. Das sind die Werkzeuge im Magazin und die entladenen Werkzeuge, deren Daten vorgehalten werden sollen. Die Werkzeugverwaltung arbeitet mit den beladenen Werkzeugen aus der Werkzeugliste.

Magazinliste

Die Magazinliste ist ein platzorientiertes Abbild des Werkzeugmagazins, der Greifer und der Spindel. Die Werkzeugverwaltung arbeitet nur mit den Werkzeugen aus der Magazinliste. Beim Werkzeugwechsel können aber auch zusätzliche Werkzeuge ohne Magazinzuordnung ausgewählt werden. Das Werkzeug muss von Hand in die Maschine eingesetzt werden und nach der Bearbeitung wieder von Hand entnommen werden (Handwerkzeug).

Belademagazin

Das Belademagazin ist das 1. interne Magazin und wird mit der Magazinnummer 9999 geführt. Im Belademagazin sind Beladestellen für das Beladen und Entladen von Werkzeugen enthalten.

Bei der Vergabe der Plätze gibt es eine feste Belegung, alle anderen Plätze sind frei vorgebbar. Bei der festen Belegung handelt es sich um den Platz 1 innerhalb des Belademagazins. Der Platz 1 ist für das Beladen/Entladen in alle Spindeln/Werkzeughalter reserviert.

Weiterhin werden alle Positionieraufträge und Umspeicheraufträge zu beliebigen Plätzen (nicht Beladestellen) über Platz 1 abgewickelt. Die genannten Aufträge, die sich auf einen bestimmten Beladeplatz beziehen, werden an der Schnittstelle dieses Beladeplatzes ausgegeben. Die Zuordnung der Beladeplätze zu Magazinen erfolgt bei der Inbetriebnahme ($TC_MDP1). Eine Beladestelle ist ein offener Zugang zum Magazin, an dem das Werkzeug manuell direkt in das Magazin ein- oder auswechselt wird.

Zwischenspeicher

Beim Zwischenspeicher handelt es sich um das 2. interne Magazin. Unter Zwischenspeicher sind Spindel, Werkzeughalter, Greifer, Lader und Übergabeplatz zusammengefasst. Die Zwischenspeicher werden unter der Magazinnummer 9998 geführt. Jedem Zwischenspeicher-Element wird ein eindeutiger Platz zugeordnet. Die Vergabe der Platznummer kann beliebig erfolgen. Es wird empfohlen, alle Spindeln oder Werkzeughalter ab der Nummer 1 aufwärts zu nummerieren. Die Zuordnung zu realen Magazinen oder zwischen Spindel/Werkzeughalter zu anderen Zwischenspeichern wird bei der Inbetriebnahme vorgenommen ($TC_MDP2, $TC_MLSR).

Kettenmagazine

Die Einstellung im MD22550: $MC_TOOL_CHANGE_MODE darf für diese Magazinarten nur den Wert 1 haben.

Bei Kettenmagazinen sind in der Regel zusätzliche Zwischenspeicher zum Transport zwischen Magazin und Spindel vorhanden. In diesen zusätzlichen Zwischenspeichern können sich temporär Werkzeuge befinden.

Bezeichnung der Zwischenspeicher und Beladestellen:

Magazin	Platz	Bedeutung
1	xx	reales Magazin 1 (Kette, Teller, Fläche), Position xx
9998	1	Spindel
9998	2	Greifer
9998	3	Greifer
9998	4	Toolboy
9998	5	Shifter
9999	1	Beladestelle für Spindel, Handwerkzeug
9999	2	Beladestelle für Magazin

Revolvermagazin

Die Einstellung von MD22550: $MC_TOOL_CHANGE_MODE hat üblicherweise den Wert 0.

Bei Revolvermagazinen sind keine zusätzlichen Zwischenspeicher vorhanden um einen Transport von Magazin nach Spindel durchzuführen. Die Werkzeuge bei Revolvermagazinen werden physikalisch nicht in die Spindel transportiert. Sie werden in eine definierte Lage durch Verdrehen des Revolvers bewegt, um die Bearbeitung mit einem Werkzeug durchzuführen. Der Transport in die Spindel oder den Werkzeughalter wird nur datenmäßig durchgeführt. Ein Transport des Werkzeugs nach dem Zwischenspeicher 9998/1 (Spindel) dient dazu der Werkzeugverwaltung mitzuteilen das der Revolver mit dem angeforderten Werkzeug zur Bearbeitungsposition gedreht wurde.

Der Programmierbefehl T = Bezeichner löst den Werkzeugwechsel aus. Alternativ kann auch T = Platz programmiert werden. Bei T = Platz ist es zulässig, dass sich kein Werkzeug auf diesem Platz befindet.

Bezeichnung der Zwischenspeicher und Beladestellen:

Magazin	Platz	Bedeutung
1	xx	reales Magazin 1 (Revolver), Position xx
9998	1	Werkzeughalter
9999	1	Beladestelle für Werkzeughalter, Handwerkzeug

Falls auch für den Revolver in MD22550: $MC_TOOL_CHANGE_MODE der Wert 1 eingestellt ist, gilt die Aussage wie bei Kettenmagazinen.

Nebenplatzbetrachtung

Eine Nebenplatzbetrachtung ist bei der Verwendung von übergroßen Werkzeugen von Bedeutung. Bei der Leerplatzsuche (Beladen, Wechsel) werden dann im Magazinplatz Parameter $TC_MPP4 die Bits 4 ... 11 ausgewertet (Halbplatz belegt/reserviert).

Siehe auch

Weitere Hinweise finden Sie in Kapitel Maschinendaten für die Werkzeugverwaltung (Seite 328).

10.1.3 Werkzeuge von Hand beladen und entladen

Handwerkzeuge

Das Bit 1 von MD22562: $MC_TOOL_CHANGE_ERROR_MODE entscheidet, ob beim Werkzeugwechsel zusätzliche Werkzeuge ohne Magazinzuordnung ausgewählt werden können. Das automatisch ausgewählte Werkzeug muss von Hand in die Maschine eingesetzt und nach der Bearbeitung wieder von Hand entnommen werden.

Verantwortung des Bedieners

Der Bediener hat dafür zu sorgen, dass der Datensatz des Werkzeugs, das er auf die Spindel steckt, sich im NCK befindet, oder dass er zu dem im NCK abgelegten Datensatz das passende Werkzeug auf die Spindel bringt. Solche von Hand in die Bearbeitung eingebrachten Werkzeuge werden als "Handwerkzeuge" bezeichnet.

Hinweis

Für die Einhaltung der Sicherheitsvorschriften muss der Anwender über das PLC-Programm sorgen.

Sie erhalten zu einem Werkzeugwechsel, bei dem ein Handwerkzeug beteiligt ist, immer einen Alarm (17212, 17214 oder 17216) als Hinweis. Der Alarm wird durch die Werkzeugwechsel-Quittung des PLC-Anwenderprogramms gelöscht.

Zu den Handwerkzeugen zählen folgende Arten von Werkzeugen:

- Übergroße Werkzeuge
- Werkzeuge, die nicht im Magazin aufbewahrt werden können.
- Werkzeuge, die nicht über das Greifersystem der Anlage laufen dürfen.

10.2 Anwendernahtstelle NC - PLC

Übersicht

Die Werkzeugverwaltung erhält Aufträge zur Vorbereitung und Durchführung eines Werkzeugwechsels (T-Befehl, M06), einer Werkzeugbewegung (MVTOOL) oder einer Magazinpositionierung (POSM) vom Teileprogramm oder vom HMI. Aus diesen Aufträgen ermittelt die Werkzeugverwaltung die notwendigen Platzwechsel der Werkzeuge und beauftragt damit die PLC.

Programm-Komponenten und Nahtstellen:

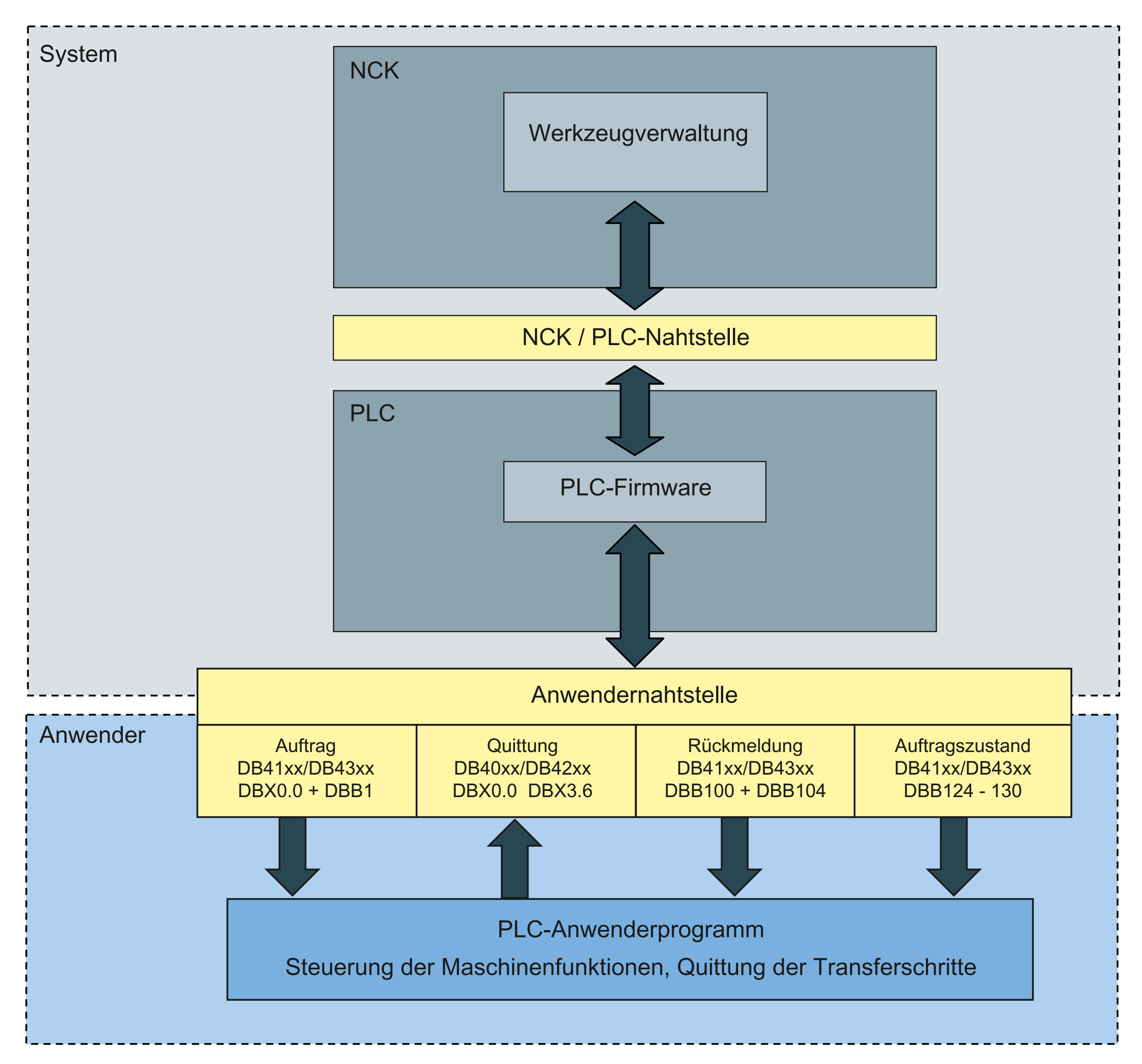


Bild 10-1 Nahtstellen der Werkzeugverwaltung

Die Anwendernahtstelle stellt getrennte Datenbausteine für Entladen, Beladen, Umsetzen und Magazin positionieren einerseits und Werkzeugwechsel andererseits zur Verfügung.

10.2.1 Werkzeug umsetzen, entladen, beladen, Magazin positionieren

Werkzeug- und Magazinbewegungen

Pro Beladestelle gibt es je eine Schnittstelle für:

- Aufträge zum Beladen, Entladen, Umsetzen von Werkzeug (MVTOOL) und zum Magazinpositionieren (POSM).

  Meldung an das PLC-Anwenderprogramm:

  – Auftrag aktiv

  – Auftragsspezifikation

  – Auftragsbeschreibung

  Die Aufträge erscheinen in der Schnittstelle der Beladestelle, von der aus ein Werkzeug eingewechselt oder wohin ein Werkzeug ausgewechselt werden soll.

- Quittungen des PLC-Anwenderprogramms

  Alle Quittungen zu einem Auftrag müssen in der Schnittstelle der gleichen Beladestelle erfolgen. Quittungsfehler sind ebenfalls in dieser Schnittstelle zurückzusetzen.

- Rückmeldung der Werkzeugverwaltung an das PLC-Anwenderprogramm

  Meldung an das PLC-Anwenderprogramm:

  – Zustand der Quittung

  – Fehlerstatus

  – Abbild der Quittungsbits

- Auftragszustand

  Ausgewählte Daten der letzten Zwischen- oder Ende-Quittung werden gespeichert. Diese Daten werden von der PLC-Firmware für die nächste Quittung an die Werkzeugverwaltung benötigt und sind zu Diagnosezwecken lesbar. Vom PLC-Anwenderprogramm können diese Daten für das Wiederaufsetzen nach einem Abbruch (z. B. Reset während eines Werkzeugwechsels) genutzt werden.

Regel

Die Verteilung der Aufträge auf die Schnittstellen erfolgt nach folgender Regel:

Enthält der Auftrag eine Beladestelle (9999/x), wird deren Schnittstelle benutzt.

Andernfalls wird die Schnittstelle der 1. Beladestelle (9999/1) benutzt.

Alle Quittungen zu einem Auftrag müssen in der Schnittstelle der gleichen Beladestelle erfolgen.

Nahtstellensignal	Bedeutung
xx: Beladestelle	
DB40xx.DBX0.0 – DBX 3.6	PLC-Anwenderprogramm: Quittungen zum Werkzeug Be-/Entladen, Umsetzen oder zum Positionieren des Magazins
DB40xx.DBX9.0	PLC-Anwenderprogramm: Rücksetzen der Meldung "Quittung fehlerhaft" (DB41xx.DBX100.1) und der Diagnose-Informationen in der Rückmeldeschnittstelle
DB41xx.DBX0.0	Werkzeugverwaltung: Auftrag zum Werkzeug Be-/Entladen, Umsetzen oder zum Positionieren des Magazins
DB41xx.DBB1	Werkzeugverwaltung: Auftragsspezifikation
DB41xx.DBW 6 – DBW34	Auftragsbeschreibung
DB41xx.DBX100:0 DB41xx.DBX100.1	Positive Rückmeldung: Quittungszustand, Quittung OK, 1 PLC-Zyklus anstehend Negative Rückmeldung: Quittungszustand, Quittung fehlerhaft, statisch anstehend
DB41xx.DBB104	Werkzeugverwaltung: Rückmeldung Fehlerstatus
DB41xx.DBX108.0 - DB41xx.DBX111.6	Abbild der Quittungen zum Werkzeug beladen, entladen, umsetzen oder zum Magazin positionieren. Dieses Abbild gehört zur positiven oder negativen Rückmeldung und ist so lange wie diese gültig.
DB41xx.DBW124 – DBW130	Auftragszustand

Aufträge

DB4100...41xx	Signale von Werkzeugverwaltung [r]							
xx: Beladestelle								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0								Auftrag
DBB1				Auftrag von NC-Programm	Positio-nieren	Umsetzen	Entladen	Beladen
DBB2	reserviert							
DBB3	reserviert							
DBB4	reserviert							
DBB5	reserviert							
DBW6	Quell-Magazinnummer (INT)							
DBW8	Quell-Platznummer (INT)							
DBW10	Ziel-Magazinnummer (INT)							

DB4100...41xx	Signale von Werkzeugverwaltung [r]							
DBW12	Ziel-Platznummer (INT)							
DBW14: HMI → PLC								Beladen/Entladen ohne Magazinbewegung

Signalbeschreibung:

- Auftrag:

  Die Schnittstelle enthält einen Auftrag. Die Auftragsbearbeitung wurde noch nicht mit einer Ende-Quittung abgeschlossen. Nach Übertragung der Ende-Quittung an die Werkzeugverwaltung wird dieses Signal rückgesetzt.
- Beladen:

  Über die Beladestelle des Parameters "Quellplatz" soll der Magazinplatz des Parameters "Zielplatz" mit einem Werkzeug beladen werden.
- Entladen:

  Das Werkzeug im Magazinplatz des Parameters "Quellplatz" soll nach der Entladestelle des Parameters "Zielplatz" entladen werden.
- Umsetzen:

  Das Werkzeug im Magazinplatz des Parameters "Quellplatz" soll nach dem Magazinplatz des Parameters "Zielplatz" umgesetzt werden.
- Positionieren:

  Der Magazinplatz des Parameters "Quellplatz" soll an die Wechsel-/Be-/Entladestelle des Parameters "Zielplatz" positioniert werden. Das Werkzeug verbleibt in seinem Magazinplatz.
- NC-Programm positioniert Magazin:

  Der Positionierauftrag kommt vom Teileprogramm.
- Beladen/Entladen ohne Magazinbewegung:

  Das Signal wird bei Einlesen von Rüstdaten oder der Funktion "alle Werkzeuge entladen" zusätzlich zu den Aufträgen für Be- oder Entladen vom HMI gesetzt. Das Signal kann im PLC-Anwenderprogramm zur Verriegelung unerwünschter Magazinbewegungen verwendet werden (z. B. nur Totalquittung setzen, keine weitere Aktion). Das Signal wird mit gültiger Ende-Quittung des Auftrags an die Werkzeugverwaltung vom System zurückgesetzt.
- Quellplatz:

  Magazin- und Platznummer eines Werkzeugs das fortbewegt oder an eine Wechsel- oder Beladstelle positioniert werden soll.
- Zielplatz:

  Magazin- und Platznummer wohin ein Werkzeug bewegt oder ein Magazinplatz positioniert werden soll.

Quittungen

DB4000...40xx	Signale an Werkzeugverwaltung [r/w]							
xx: Beladestelle								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0	Quittungs-schritt 7	Quittungs-schritt 6	Quittungs-schritt 5	Quittungs-schritt 4	Quittungs-schritt 3	Quittungs-schritt 2	Quittungs-schritt 1	Total-Quittung
DBB1	Quittungs-schritt 15	Quittungs-schritt 14	Quittungs-schritt 13	Quittungs-schritt 12	Quittungs-schritt 11	Quittungs-schritt 10	Quittungs-schritt 9	Quittungs-schritt 8
DBB2	Quittungs-schritt 23	Quittungs-schritt 22	Quittungs-schritt 21	Quittungs-schritt 20	Quittungs-schritt 19	Quittungs-schritt 18	Quittungs-schritt 17	Quittungs-schritt 16
DBB3	reserviert	Quittungs-schritt 30	Quittungs-schritt 29	Quittungs-schritt 28	Quittungs-schritt 27	Quittungs-schritt 26	Quittungs-schritt 25	Quittungs-schritt 24
DBB4	reserviert							
DBB5	reserviert							
DBB6	reserviert							
DBB7	reserviert							
DBB8	reserviert							
DBB9								Rücksetzen Quittungsfehler

Signalbeschreibung:

- Total-Quittung:

  Bei einer 0/1-Flanke wird zum aktuellen Auftrag die Ende-Quittung mit Status 99 gesendet (Auftrag komplett erledigt, alle Ziel-Positionen sind erreicht). Solange das Signal ansteht, dürfen keine Änderungen an den Daten dieser Schnittstelle vorgenommen werden!

  Nach der Übertragung der Quittung an die Werkzeugverwaltung wird dieses Signal von der PLC-Firmware zurückgesetzt.

- Quittungsschritt 1…30:

  Bei einer 0/1-Flanke wird der zugehörige Quittungsschritt aus der Quittungsschritt-Tabelle an die Werkzeugverwaltung gesendet. Solange das Signal ansteht, dürfen keine Änderungen an den Daten dieser Schnittstelle und der variablen Transferschritt-Tabelle vorgenommen werden!

  Nach der Übertragung der Quittung an die Werkzeugverwaltung wird dieses Signal von der PLC-Firmware zurückgesetzt.

- Rücksetzen Quittungsfehler:

  Rücksetzen der Meldung "Quittung fehlerhaft" (DB41xx.DBX100.1) und der Diagnose-Informationen in der Rückmeldeschnittstelle.

Rückmeldungen

DB4100...41xx	Signale von Werkzeugverwaltung [r]							
xx: Beladestelle								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB100							Quittung fehlerhaft	Quittung OK
DBB101	reserviert							
DBB102	reserviert							
DBB103	reserviert							
DBW104	Fehlerstatus (WORD)							
DBB106	reserviert							
DBB107	reserviert							
DBB108	Quittungs-schritt 7	Quittungs-schritt 6	Quittungs-schritt 5	Quittungs-schritt 4	Quittungs-schritt 3	Quittungs-schritt 2	Quittungs-schritt 1	Quittungs-schritt 0
DBB109	Quittungs-schritt 15	Quittungs-schritt 14	Quittungs-schritt 13	Quittungs-schritt 12	Quittungs-schritt 11	Quittungs-schritt 10	Quittungs-schritt 9	Quittungs-schritt 8
DBB110	Quittungs-schritt 22	Quittungs-schritt 22	Quittungs-schritt 21	Quittungs-schritt 20	Quittungs-schritt 19	Quittungs-schritt 18	Quittungs-schritt 17	Quittungs-schritt 16
DBB111	reserviert	Quittungs-schritt 30	Quittungs-schritt 29	Quittungs-schritt 28	Quittungs-schritt 27	Quittungs-schritt 26	Quittungs-schritt 25	Quittungs-schritt 24

Signalbeschreibung:

- **Quittung OK (DB41xx.DBX100.0):**

  Die Quittung des PLC-Anwenderprogramm (Bereich DB40xx.DBB0 bis DBB3) wurde fehlerfrei an die Werkzeugverwaltung übertragen. Dieses Signal wird nach einem PLC-Takt rückgesetzt.

- **Quittung fehlerhaft (DB41xx.DBX100.1):**

  Negative Rückmeldung Quittungszustand. Die Quittung des PLC-Anwenderprogramm (Bereich DB40xx.DBB0 bis DBB3) ist fehlerhaft. Die Fehlerursache wird in "Fehlerstatus" mitgeteilt.

  Das Bit "Quittung fehlerhaft" wird auch gesetzt, wenn die Quittung von der PLC-Firmware als fehlerfrei akzeptiert wurde und die Werkzeugverwaltung einen Fehler im quittierten Werkzeugtransfer erkennt und diesen meldet (z. B. wenn der Zielplatz des Werkzeugtransfers belegt ist).

  Fehler, die direkt von der Werkzeugverwaltung in der NC erkannt werden, ohne dass vom PLC-Anwenderprogramm eine Quittung gegeben wurde, führen nicht zum Setzen des Bits 100.1.

  Liegt ein Fehler vor, der die Weitergabe der Quittung an die Werkzeugverwaltung verhindert (Fehlerstatus 1 bis 7), wird der Fehler nur an der Nahtstelle der Werkzeugverwaltung und nicht von der NC ausgegeben (kein NC Alarm!).

  Bei Bedarf melden Sie diese Fehler mit einem Anwender PLC-Alarm durch das PLC-Anwenderprogramm.

Dieses Signal bleibt bis zum Quittieren des Fehlers statisch anstehen (Setzen des Bits "Rücksetzen Quittungsfehler" DB40xx.DBX9.0) durch den Anwender. Die Nahtstelle in DB40xx.DBB0 bis DBB3 ist bei anstehendem Bit "Quittung fehlerhaft" gesperrt. Eingehende Quittungsbits werden von der PLC-Firmware nicht ausgewertet und mit dem Setzen des Bits "Rücksetzen Quittungsfehler" gelöscht.

Fehlerstatus:

Der Fehlerstatus (DB41xx.DBB104) enthält bei einem Fehler eine Diagnose-Nummer ungleich Null.

Status	Bedeutung
0	kein Fehler
1	mehrere Quittungssignale gleichzeitig
2	Quittung ohne Auftrag
3	ungültige Transferschritt-Nummer
4	für eine Positionsangabe fehlt der Auftrag
5	Status erlaubt keine Platzänderung (Quittungsstatus 0 wurde verwendet)
7	Ein unerlaubter Quittungsstatus wurde verwendet
andere Werte:	Die Zahl entspricht der Fehlermeldung der Werkzeugverwaltung im NCK, die durch diesen Transfer verursacht wurde

Der Fehlerstatus wird mit dem Quittieren des Fehlers durch den Anwender zurückgesetzt.

Abbild der Quittungen (DB41xx.DBB108 bis DBB111)

Die vom PLC-Anwenderprogramm zuletzt gesetzten Quittungen (DB40xx.DBB0 bis DBB3) werden hier von der PLC-Firmware zusammen mit den Bits "Quittung OK" oder "Quittung fehlerhaft" gesetzt und rückgesetzt. Anhand dieser im Fehlerfall statisch anstehenden Bits erkennt der Anwender, welcher Quittungsschritt den Fehler ausgelöst hat. Wurden unrichtigerweise vom PLC-Anwenderprogramm mehrere Quittungsbits gesetzt, werden auch diese in das Abbild eingetragen.

Auftragszustand

DB4100...41xx	Signale von Werkzeugverwaltung [r]							
xx: Beladestelle								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW124	aktuelle Magazinnummer des Werkzeugs (INT)							
DBW126	aktuelle Platznummer des Werkzeugs (INT)							
DBW128	Ziel-Magazinnummer des Werkzeugs (INT)							
DBW130	Ziel-Platznummer des Werkzeugs (INT)							

10.2.2 Werkzeugwechsel

Nahtstellenbeschreibung

Pro Werkzeughalter/Spindel gibt es je eine Schnittstelle für:

- Aufträge zur Vorbereitung und Durchführung des Werkzeugwechsels.

  Meldung an das PLC-Anwenderprogramm: Auftrag aktiv; Auftragsspezifikation und Auftragsbeschreibung.

  Die Aufträge erscheinen in der Schnittstelle des Werkzeughalters (der Spindel), woher ein Werkzeug eingewechselt oder aus dem ein Werkzeug ausgewechselt werden soll.
- Quittungen des PLC-Anwenderprogramms

  Alle Quittungen zu einem Auftrag müssen in der Schnittstelle des gleichen Werkzeughalters (der gleichen Spindel) erfolgen. Quittungsfehler sind ebenfalls in dieser Schnittstelle zurückzusetzen.
- Rückmeldung der Werkzeugverwaltung an das PLC-Anwenderprogramm

  Meldung an das PLC-Anwenderprogramm: Zustand der Quittung, Fehlerstatus, Abbild der Quittungsbits.
- Auftragszustand

  Ausgewählte Daten der letzten Zwischen- oder Ende-Quittung werden gespeichert. Diese Daten werden von der PLC-Firmware für die nächste Quittung an die Werkzeugverwaltung benötigt und sind zu Diagnosezwecken lesbar. Vom PLC-Anwenderprogramm können diese Daten für das Wiederaufsetzen nach einem Abbruch (z. B. Reset während eines Werkzeugwechsels) genutzt werden.

Ende-Quittung für Werkzeugwechsel

Für "Werkzeugwechsel vorbereiten" und "Werkzeugwechsel durchführen" kann es eine gemeinsame (Werkzeugwechsel durch T-Befehl bei Revolver) oder getrennte (Txx und M206 in verschiedenen Sätzen, Voreinstellung Fräsen) Ende-Quittungen geben. Die Ende-Quittung auf den Auftrag zum Werkzeugwechsel vorbereiten, lässt bei entsprechender MD-Einstellung den Vorlauf des NCK weiterlaufen.

Die Maschinendaten mit denen das Verhalten von Satzvorlauf, Hauptlauf und das unterschiedliche Quittungsverhalten sind in Kapitel Einstellungen für die Werkzeugverwaltung (Seite 328) beschrieben.

Mit der Ende-Quittung auf den Auftrag "Werkzeugwechsel durchführen" kann der Hauptlauf des NCK fortgesetzt werden. Diese Ende-Quittungen sollten deshalb so früh wie möglich erfolgen. Das kann bedeuten, dass die Ende-Quittung erfolgt, bevor das Alt-Werkzeug im Magazin ist (z. B. das Neu-Werkzeug befindet sich in der Spindel, das Alt-Werkzeug im Toolboy). Die restlichen Schritte des Alt-Werkzeugs bis ins Magazin müssen dann asynchron mitgeteilt werden. Dazu ist die gleiche Schnittstelle wie für die synchronen Quittungen zu verwenden.

Nahtstellensignal	Bedeutung
xx: Index der Spindel / Werkzeughalter	
DB42xx.DBX0.0 – DBX 3.6	PLC-Anwenderprogramm: Quittungen zum Werkzeugwechsel vorbereiten und durchführen
DB42xx.DBX9.0	PLC-Anwenderprogramm: Rücksetzen der Meldung "Quittung fehlerhaft" (DB43xx.DBX100.1) und der Diagnose-Informationen in der Rückmeldungsschnittstelle
DB43xx.DBX0.0	Werkzeugverwaltung: Auftrag zum "Werkzeugwechsel vorbereiten" und "Werkzeugwechsel durchführen"
DB43xx.DBB1	Werkzeugverwaltung Auftragsspezifikation
DB43xx.DBW 6 – DBW34	Auftragsbeschreibung
DB43xx.DBX100.0 DB43xx.DBX100.1	Positive Rückmeldung Quittungszustand, Quittung OK, 1 PLC-Zyklus anstehend Negative Rückmeldung Quittungszustand, Quittung fehlerhaft, statisch anstehend
DB43xx.DBX100:0 DB43xx.DBX100:1	Positive Rückmeldung Quittungszustand, 1 PLC-Zyklus anstehend Negative Rückmeldung Quittungszustand, statisch anstehend.
DB43xx.DBX100:0	Werkzeugverwaltung Rückmeldung Quittungszustand
DB43xx.DBB104	Werkzeugverwaltung Rückmeldung Fehlerstatus
DB43xx.DBX108.0 - DB43xx.DBX111.6	Abbild der Quittungen zum Werkzeugwechsel: Dieses Abbild gehört zur positiven oder negativen Rückmeldung und ist ebenso lange gültig.
DB43xx.DBW124 – DBW138	Auftragszustand

Aufträge

DB4300...43xx	Signale von Werkzeugverwaltung [r]							
xx: Werkzeughalter								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0	--	--	--	--	--	--	--	Auftrag
DBB1	Werkzeug verbleibt in der Spindel	Hand-werkzeug aus-wechseln	Hand-werkzeug ein-wechseln	kein Alt-Werkzeug	T0	Wechsel vor-bereiten	Wechsel durch-führen (Anstoß: M06)	festplatz-codiert
DBB2	reserviert							
DBB3	reserviert							
DBB4	reserviert							
DBB5	reserviert							
DBW6	Quell-Magazinnummer für neues Werkzeug (INT)							
DBW8	Quell-Platznummer für neues Werkzeug (INT)							
DBW10	reserviert							
DBW12	reserviert							

<table>
<tr><th>DB4300...43xx</th><th colspan="8">Signale von Werkzeugverwaltung [r]</th></tr>
<tr><td>DBW14</td><td colspan="8">reserviert</td></tr>
<tr><td>DBW16</td><td colspan="8">reserviert</td></tr>
<tr><td>DBW18</td><td colspan="8">Ziel-Magazinnummer für altes Werkzeug (INT)</td></tr>
<tr><td>DBW20</td><td colspan="8">Ziel-Platznummer für altes Werkzeug (INT)</td></tr>
<tr><td>DBW22</td><td colspan="8">Platztyp (INT)</td></tr>
<tr><td>DBW24</td><td colspan="8">Größe links (INT)</td></tr>
<tr><td>DBW26</td><td colspan="8">Größe rechts (INT)</td></tr>
<tr><td>DBW28</td><td colspan="8">reserviert</td></tr>
<tr><td>DBW30</td><td colspan="8">reserviert</td></tr>
<tr><td rowspan="2">DBB32</td><td colspan="8">Werkzeugstatus für neues Werkzeug:</td></tr>
<tr><td>Werkzeug war im Einsatz</td><td>Werkzeug festplatz-codiert</td><td>Werkzeug im Wechsel</td><td>Vorwarn-grenze erreicht</td><td>Werkzeug ver-messen</td><td>Werkzeug gesperrt</td><td>Werkzeug frei-gegeben</td><td>aktives Werkzeug</td></tr>
<tr><td rowspan="2">DBB33</td><td colspan="8">Werkzeugstatus für neues Werkzeug:</td></tr>
<tr><td>Handwerk zeug</td><td>1:1-Tausch</td><td>--</td><td>Stamm-werkzeug</td><td>zu beladen</td><td>zu entladen</td><td>gesperrt</td><td>Kennung für Werkzeug e im Zwischen-speicher</td></tr>
<tr><td>DBW34</td><td colspan="8">Neues Werkzeug: Interne T-Nummer des NCK (INT)</td></tr>
<tr><td>DBW36</td><td colspan="8">reserviert</td></tr>
<tr><td>DBW38</td><td colspan="8">reserviert</td></tr>
<tr><td>DBW40</td><td colspan="8">reserviert</td></tr>
<tr><td>DBW42</td><td colspan="8">reserviert</td></tr>
<tr><td>DBW44</td><td colspan="8">Freiparameter 1 (DWORD)</td></tr>
<tr><td>DBW48</td><td colspan="8">Freiparameter 2 (DWORD)</td></tr>
<tr><td>DBW52</td><td colspan="8">Freiparameter 3 (DWORD)</td></tr>
</table>

Signalbeschreibung:

- Auftrag:

  Die Schnittstelle enthält einen Auftrag. Die Auftragsbearbeitung wurde noch nicht mit einer Ende-Quittung abgeschlossen. Nach Übertragung der Ende-Quittung an die Werkzeugverwaltung wird dieses Signal rückgesetzt.

- Festplatz-codiert: Das neue Werkzeug ist festplatz-codiert.

- Werkzeugwechsel durchführen:

  Das neue Werkzeug ist in den Werkzeughalter / die Spindel einzuwechseln. Das alte Werkzeug ist auf einen Magazinplatz zurückzubringen. Dieser Auftrag erfordert immer eine Ende-Quittung.

- Werkzeugwechsel vorbereiten:

  Neues Werkzeug bereitstellen. Magazinplatz evtl. für altes Werkzeug an die Wechselstelle positionieren. Dieser Auftrag erfordert einzeln eine Ende-Quittung, bei gleichzeitigem Auftrag "Wechsel durchführen" entfällt die Ende-Quittung auf die Vorbereitung.

- T0: T0 wurde programmiert (Werkzeughalter/Spindel leer fahren).
- kein Alt-Werkzeug:

  Werkzeugwechsel in den bisher leeren Werkzeughalter/Spindel.
- Handwerkzeug einwechseln:

  Ein Handwerkzeug ist einzuwechseln. Welches Werkzeug eingewechselt werden soll, wird am HMI angezeigt.
- Handwerkzeug auswechseln:

  Das Werkzeug ist über Handbedienung auszuwechseln.
- Werkzeug verbleibt in Spindel:

  Das Bit wird gesetzt bei einem Wechsel von Werkzeughalter → Spindel nach Werkzeughalter → Spindel. Auslöser kann z. B. Reset- Startmode oder Satzsuchlauf sein.
- Quellplatz neues Werkzeug:

  Magazin- und Platznummer, woher das neue Werkzeug kommt (meist ein Platz in einem realen Magazin).
- Zielplatz altes Werkzeug:

  Magazin- und Platznummer, wohin das alte Werkzeug zu transportieren ist (meist ein Platz in einem realen Magazin).
- Ursprung neues Werkzeug:
  - Interne T-Nummer: Interne T-Nummer des neuen Werkzeugs
  - Werkzeugstatus: Werkzeugstatus des neuen Werkzeugs
  - Platztyp: Platztyp des neuen Werkzeugs
  - Größe: Größe (rechts, links, oben, unten) des neuen Werkzeugs
  - Freie Parameter: 3 freie Parameter, die vom Teileprogramm an das PLC-Anwenderprogramm mitgegeben werden.

Quittungen

DB4200 ... 42xx	Signale an Werkzeugverwaltung [r/w]							
xx: Werkzeughalter								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0	Quittungs-schritt 7	Quittungs-schritt 6	Quittungs-schritt 5	Quittungs-schritt 4	Quittungs-schritt 3	Quittungs-schritt 2	Quittungs-schritt 1	Total-Quittung
DBB1	Quittungs-schritt 15	Quittungsschritt 14	Quittungs-schritt 13	Quittungs-schritt 12	Quittungs-schritt 11	Quittungs-schritt 10	Quittungs-schritt 9	Quittungs-schritt 8
DBB2	Quittungs-schritt 23	Quittungs-schritt 22	Quittungs-schritt 21	Quittungs-schritt 20	Quittungs-schritt 19	Quittungs-schritt 18	Quittungs-schritt 17	Quittungs-schritt 16
DBB3	reserviert	Quittungs-schritt 30	Quittungs-schritt 29	Quittungs-schritt 28	Quittungs-schritt 27	Quittungs-schritt 26	Quittungs-schritt 25	Quittungs-schritt 24
DBB4	reserviert							
DBB5	reserviert							

DB4200 ... 42xx	Signale an Werkzeugverwaltung [r/w]							
DBB6	reserviert							
DBB7	reserviert							
DBB8	reserviert							
DBB9								Rücksetzen Quittungs-fehler

Signalbeschreibung:

- Total-Quittung:

  Bei einer 0/1-Flanke wird zum aktuellen Auftrag die Ende-Quittung mit Status 99 gesendet (Auftrag komplett erledigt, alle Ziel-Positionen sind erreicht). Solange das Signal ansteht, dürfen keine Änderungen an den Daten dieser Schnittstelle vorgenommen werden!

  Nach der Übertragung der Quittung an die Werkzeugverwaltung wird dieses Signal von der PLC-Firmware zurückgesetzt.

- Quittungsschritt 1…30:

  Bei einer 0/1-Flanke wird der zugehörige Quittungsschritt aus der Quittungsschritt-Tabelle an die Werkzeugverwaltung gesendet. Solange das Signal ansteht, dürfen keine Änderungen an den Daten dieser Schnittstelle und der variablen Transferschritt- Tabelle vorgenommen werden!

  Nach der Übertragung der Quittung an die Werkzeugverwaltung wird dieses Signal von der PLC-Firmware zurückgesetzt.

- Rücksetzen Quittungsfehler:

  Rücksetzen der Meldung Quittung fehlerhaft (DB43xx.DBX100.1) und der Diagnose-Informationen in der Rückmeldeschnittstelle.

Rückmeldungen

DB4300...43xx	Signale von Werkzeugverwaltung [r]							
xx: Beladestelle								
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB100							Quittung fehlerhaft	Quittung OK
DBB101	reserviert							
DBB102	reserviert							
DBB103	reserviert							
DBW104	Fehlerstatus (WORD)							
DBB106	reserviert							
DBB107	reserviert							
DBB108	Quittungs-schritt 7	Quittungs-schritt 6	Quittungs-schritt 5	Quittungs-schritt 4	Quittungs-schritt 3	Quittungs-schritt 2	Quittungs-schritt 1	Quittungs-schritt 0
DBB109	Quittungs-schritt 15	Quittungs-schritt 14	Quittungs-schritt 13	Quittungs-schritt 12	Quittungs-schritt 11	Quittungs-schritt 10	Quittungs-schritt 9	Quittungs-schritt 8

DB4300...43xx	Signale von Werkzeugverwaltung [r]							
DBB110	Quittungs-schritt 23	Quittungs-schritt 22	Quittungs-schritt 21	Quittungs-schritt 20	Quittungs-schritt 19	Quittungs-schritt 18	Quittungs-schritt 17	Quittungs-schritt 16
DBB111	reserviert	Quittungs-schritt 30	Quittungs-schritt 29	Quittungs-schritt 28	Quittungs-schritt 27	Quittungs-schritt 26	Quittungs-schritt 25	Quittungs-schritt 24

Signalbeschreibung:

- **Quittung OK (DB43xx.DBX100.0): Positive Rückmeldung Quittungszustand.**

  Die Quittung des PLC-Anwenderprogramm (Bereich DB42xx.DBB0 bis DBB3) wurde fehlerfrei an die Werkzeugverwaltung übertragen. Dieses Signal wird nach einem PLC-Takt rückgesetzt.

- **Quittung fehlerhaft (DB43xx.DBX100.1): Negative Rückmeldung Quittungszustand.**

  Die Quittung des PLC-Anwenderprogramm (Bereich DB42xx.DBB0 bis DBB3) ist fehlerhaft. Die Fehlerursache wird in "Fehlerstatus" mitgeteilt.

  Das Bit "Quittung fehlerhaft" wird auch gesetzt wenn die Quittung von der PLC-Firmware als fehlerfrei akzeptiert wurde und die Werkzeugverwaltung einen Fehler im quittierten Werkzeugtransfer erkennt und diesen meldet (zum Beispiel wenn der Zielplatz des Werkzeugtransfers belegt ist).

  Fehler die direkt von der Werkzeugverwaltung in der NC erkannt werden, ohne das vom PLC-Anwenderprogramm eine Quittung gegeben wurde, führen nicht zum Setzen des Bits 100.1.

  Liegt ein Fehler vor, der die Weitergabe der Quittung an die Werkzeugverwaltung verhindert (Fehlerstatus 1 bis 7), wird der Fehler nur an der Nahtstelle der Werkzeugverwaltung und nicht von der NC ausgegeben (kein NC Alarm!).

  Bei Bedarf melden Sie diese Fehler mit einem Anwender PLC-Alarm durch das PLC-Anwenderprogramm.

  Dieses Signal bleibt bis zum Quittieren des Fehlers statisch anstehen (Setzen des Bits "Rücksetzen Quittungsfehler" DB4200.DBX9.0) durch den Anwender. Die Nahtstelle in DB42xx.DBB0 bis DBB3 ist bei anstehendem Bit "Quittung fehlerhaft" gesperrt. Eingehende Quittungsbits werden von der PLC-Firmware nicht ausgewertet und ebenfalls mit dem Setzen des Bits "Rücksetzen Quittungsfehler" gelöscht.

Fehlerstatus:

Der Fehlerstatus (DB43xx.DBB104) enthält bei einem Fehler eine Diagnose-Nummer ungleich Null.

Status	Bedeutung
0	kein Fehler
1	mehrere Quittungssignale gleichzeitig
2	Quittung ohne Auftrag
3	ungültige Transferschritt-Nummer
4	für eine Positionsangabe fehlt der Auftrag
5	Status erlaubt keine Platzänderung (Quittungsstatus 0 wurde verwendet)
7	Ein unerlaubter Quittungsstatus wurde verwendet
andere Werte:	Die Zahl entspricht der Fehlermeldung der Werkzeugverwaltung im NCK, die durch diesen Transfer verursacht wurde.

Der Fehlerstatus wird mit dem Quittieren des Fehlers durch den Anwender zurückgesetzt.

Abbild der Quittungen (DB43xx.DBB108 bis DBB111)

Die vom PLC-Anwenderprogramm zuletzt gesetzten Quittungen (DB42xx.DBB0 bis DBB3) werden hier von der PLC-Firmware zusammen mit den Bits "Quittung OK" oder "Quittung fehlerhaft" gesetzt und rückgesetzt. Anhand dieser im Fehlerfall statisch anstehenden Bits erkennt der Anwender, welcher Quittungsschritt den Fehler ausgelöst hat. Wurden unrichtigerweise vom PLC-Anwenderprogramm mehrere Quittungsbits gesetzt, werden diese auch in das Abbild eingetragen.

Auftragszustand

DB4300 ... 43xx	Signale von Werkzeugverwaltung [r]							
xx: Werkzeughalter								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW124	aktuelle Magazinnummer für neues Werkzeug (INT)							
DBW126	aktuelle Platznummer für neues Werkzeug (INT)							
DBW128	Ziel-Magazinnummer für neues Werkzeug (INT)							
DBW130	Ziel-Platznummer für neues Werkzeug (INT)							
DBW132	aktuelle Magazinnummer für altes Werkzeug (INT)							
DBW134	aktuelle Platznummer für altes Werkzeug (INT)							
DBW136	Ziel-Magazinnummer für altes Werkzeug (INT)							
DBW138	Ziel-Platznummer für altes Werkzeug (INT)							

10.2.3 Transferschritt- und Quittungsschritt-Tabellen

Projektierbare Schritt-Tabellen

In den Datenbausteinen TM_CTS (DB9900), TM_VTS (DB9901) und TM_ACK (DB9902) stehen projektierbare Tabellen zur Verfügung, mit denen die Abläufe der Werkzeugbewegungen beschrieben werden.

Nahtstellensignal	Name	Bedeutung
DB9900	TM_CTS	konstante Transferschritt-Tabelle (projektierbar)
DB9901	TM_VTS	variable Transferschritt-Tabelle (projektierbar und von PLC-Anwenderprogramm beschreibbar)
DB9902	TM_ACK	Quittungsschritt-Tabelle (projektierbar)

Die Datenbausteine DB40xx, 41xx, 42xx und 43xx sind Systembausteine und werden automatisch von der Steuerung angelegt.

Die Datenbausteine DB9900, DB9901 und DB9902 werden vom Programming Tool unter Bibliotheken/Spezielle Datenbausteine zur Verfügung gestellt. Die Bausteine sind noch nicht mit den notwendigen Daten gefüllt und müssen vom Anwender in das PLC-Projekt kopiert und bearbeitet werden.

Transferschritt-Tabellen

Die einzelnen Werkzeugbewegungen werden als Transferschritte Werkzeug von Magazinplatz x/y nach Magazinplatz m/n definiert. Mit diesen Transferschritten können Quittungsschritte definiert werden. DB9900 enthält fest projektierte Transferschritte (konstante Transferschritt-Tabelle). DB 9901 kann vom PLC-Anwenderprogramm verändert werden, zum Beispiel für die Quittung von Zwischenschritten wie Magazinpositionen bei der Werkzeugwechselvorbereitung (variable Transferschritt-Tabelle).

DB9900	Konstante Transferschritt-Tabelle [r]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW0	Transferschritt 1 Quell-Magazinnummer (INT)							
DBW2	Transferschritt 1 Quell-Platznummer (INT)							
DBW4	Transferschritt 1 Ziel-Magazinnummer (INT)							
DBW6	Transferschritt 1 Ziel-Platznummer (INT)							
DBW8	Transferschritt 2 Quell-Magazinnummer (INT)							
DBW10	Transferschritt 2 Quell-Platznummer (INT)							
DBW12	Transferschritt 2 Ziel-Magazinnummer (INT)							
DBW14	Transferschritt 2 Ziel-Platznummer (INT)							
	…							
DBW504	Transferschritt 64 Quell-Magazinnummer (INT)							
DBW506	Transferschritt 64 Quell-Platznummer (INT)							

DB9900	Konstante Transferschritt-Tabelle [r]
DBW508	Transferschritt 64 Ziel-Magazinnummer (INT)
DBW510	Transferschritt 64 Ziel-Platznummer (INT)

DB9901	Variable Transferschritt-Tabelle [rw]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW0	Transferschritt 101 Quell-Magazinnummer (INT)							
DBW2	Transferschritt 101 Quell-Platznummer (INT)							
DBW4	Transferschritt 101 Ziel-Magazinnummer (INT)							
DBW6	Transferschritt 101 Ziel-Platznummer (INT)							
DBB 8	Transferschritt 102 Quell-Magazinnummer (INT)							
DBW10	Transferschritt 102 Quell-Platznummer (INT)							
DBW12	Transferschritt 102 Ziel-Magazinnummer (INT)							
DBW14	Transferschritt 102 Ziel-Platznummer (INT)							
	…							
DBW504	Transferschritt 164 Quell-Magazinnummer (INT)							
DBW506	Transferschritt 164 Quell-Platznummer (INT)							
DBW508	Transferschritt 164 Ziel-Magazinnummer (INT)							
DBW510	Transferschritt 164 Ziel-Platznummer (INT)							

Quittungsschritt-Tabelle

Jeder Eintrag indiziert zwei Transferschritte (für Neu- und Altwerkzeug) und gibt den erreichten zugehörigen Status an. Die Quittungsschritt-Tabelle in DB9902 wird gemeinsam für Quittungen auf der Nahtstelle der Beladestelle als auch auf der Nahtstelle des Werkzeughalters verwendet.

DB9902	Quittungsschritt-Tabelle [r]							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0	Quittungsschritt 1 Transferschritt für neues Werkzeug (BYTE)							
DBB1	Quittungsschritt 1 Transferschritt für altes Werkzeug (BYTE)							
DBB2	Quittungsschritt 1 Quittungsstatus (BYTE)							
DBB3	Quittungsschritt 1 reserviert							
DBB4	Quittungsschritt 2 Transferschritt für neues Werkzeug (BYTE)							
DBB5	Quittungsschritt 2 Transferschritt für altes Werkzeug (BYTE)							
DBB6	Quittungsschritt 2 Quittungsstatus (BYTE)							
DBB7	Quittungsschritt 2 reserviert							
	…							
DBB116	Quittungsschritt 30 Transferschritt für neues Werkzeug (BYTE)							
DBB117	Quittungsschritt 30 Transferschritt für altes Werkzeug (BYTE)							

DB9902	Quittungsschritt-Tabelle [r]
DBB118	Quittungsschritt 30 Quittungsstatus (BYTE)
DBB119	Quittungsschritt 30 reserviert

Siehe auch

PLC Programmbausteine (Seite 337)

10.3 Maschinendaten für die Werkzeugverwaltung

10.3.1 Einstellungen für die Werkzeugverwaltung

Maschinendaten (Voreinstellung)

Folgende Maschinendaten sind für die Werkzeugverwaltung bereits voreingestellt oder werden beim Hochlauf mit "default data" gesetzt: Diese Einstellungen können bei Bedarf geändert werden.

MD Nummer	Bezeichnung	Wert	
10715[0]	M_NO_FCT_CYLE	6	(M-Version)
10716[0]	M_NO_FCT_CYLCLE_NAME	L6	(M-Version)
10717	T_NO_FCT_CYLCLE_NAME	TCHANGE	(T-Version)
17500	MAXNUM_REPLACEMENT_TOOLS	0	
20124	TOOL_MANAGEMENT_TOOLHOLDER	1	
20270	CUTTING_EDGE_DEFAULT	1	
20310	TOOL_MANAGEMENT_MASK	181400F 180400F	(T-Version) (M-Version)
22550	TOOL_CHANGE_MODE	0 1	(T-Version) (M-Version)
22560	TOOL_CHANGE_MCODE	206	
22562	TOOL_CHANGE_ERROR_MODE	0	(für Handwerkzeug)

MD20270: $MC_CUTTING_EDGE_DEFAULT

Wird nach einem Werkzeugwechsel keine Schneide programmiert, so wird die Schneidennummer verwendet, die in $MC_CUTTING_EDGE_DEFAULT eingestellt ist.

MD20270: $MC_CUTTING_EDGE_DEFAULT	
	Grundstellung der Werkzeugschneide ohne Programmierung (DWORD)
> 0	Nummer der Schneide, die mit M206 angewählt wird. Die Schneidenanwahl erfolgt auch wenn anschließend eine D-Programmierung erfolgt.
= 1	Voreinstellung
= 0	Nach einem Werkzeugwechsel ist zunächst keine Schneide aktiv. Eine vor dem Werkzeugwechsel aktive Werkzeugkorrektur wird abgewählt (entspricht D0!). Die Schneidenanwahl erfolgt erst bei D-Programmierung.
= -1	Schneidennummer des alten Werkzeugs gilt auch für das neue Werkzeug
= -2	Schneidenkorrektur des alten Werkzeugs bleibt weiterhin aktiv bis D programmiert wird.

Die Einstellung des MD20270 hat Einfluss auf die Satzaufbereitung der NC. Um zu verhindern, dass mit Aufruf des Wechselbefehls ein Vorlauf-Stopp bis zur Quittung des Werkzeugwechsels eintritt, führen Sie NC-Funktionen ohne Werkzeugkorrektur im Werkzeugwechsel-Unterprogramm aus, zum Beispiel Achsen verfahren oder die Ausgabe von Hilfsfunktionen.

Beispiel:

Voraussetzung: MD20270: $MC_CUTTING_EDGE_DEFAULT= 0 oder = -2

Nach dem Aufruf M206 zum Werkzeugwechsel können die Achsen sofort weiterfahren, ohne auf die Wechselquittung zu warten und Verfahrsätze ohne Werkzeugkorrektur ausführen. Erst im Satz mit einer Korrekturanwahl (D-Nr.) wird angehalten bis der Werkzeugwechsel von der PLC als beendet gemeldet wurde.

Ablauf im Teileprogramm:

```
N10 T="Bohrer18"          ; Werkzeugwechselvorbereitung
N20 M6                    ; Aufruf des Werkzeugwechsel-Unterprogramms
Werkzeugwechsel-Unterprogramm L6:
N10 M206                  ; Werkzeugwechsel
N20 D0                    ; Korrekturabwahl
N40 Y150 M79              ; Maschinenachsen verfahren
N50 G01 D1 X10            ; Einschalten der Werkzeugkorrektur.
                          ; Prüfung, ob der Werkzeugwechsel erfolgt ist.
                          ; Der Vorlauf-Stopp wird solange aufrechterhalten bis die
                          Werkzeugwechsel-Vorbereitung abgeschlossen wurde. Der
                          Hauptlauf wartet bei N50 (D1) bis der Werkzeugwechsel
                          durchgeführt und quittiert ist.
```

MD20310: $MC_TOOL_MANAGEMENT_MASK

Einstellungen:

MD20310: $MC_TOOL_MANAGEMENT_MASK		
		Aktivierung der Werkzeugverwaltungsfunktionen
T- und M-Version:		
Bit 0	= 1	Werkzeugverwaltung aktiv: Die Funktionen der Werkzeugverwaltung sind für den aktuellen Kanal frei geschaltet.
Bit 1	= 1	Überwachungsfunktionen der Werkzeugverwaltung aktiv: Die Funktionen für die Überwachung der Werkzeuge (Standzeit und Stückzahl) werden frei geschaltet.
Bit 2	= 1	OEM-Funktionen aktiv
Bit 3	= 1	Nebenplatzbetrachtung aktiv

MD20310: $MC_TOOL_MANAGEMENT_MASK		
Bit 14	= 1	Werkzeug- und Korrekturanwahl entsprechend den Einstellungen in: MD20110 $MC_RESET_MODE_MASK MD20112 $MC_START_MODE_MASK
Bit 23	= 1	Bei Korrekturanwahl keine Synchronisation mit Hauptlauf.
Bit 24	= 1	Mit einem asynchronen Transfer mit Quittungsstatus 201 können Sie ein Werkzeug auf einen für ein anderes Werkzeug mit "reserviert für Werkzeug aus Zwischenspeicher" reservierten Platz bewegen. Diese Platzreservierung wird dazu vor der Ausführung der Bewegung entfernt ("reserviert für neu zu beladenes Werkzeug " (Bitwert="H8") bleibt wirksam).
zusätzlich nur bei T-Version:		
Bit 16	= 1	T-Platznummer ist aktiv.

Befindet sich ein gesperrtes Werkzeug auf dem programmiertem Platz, wird bei gesetzter Option "Ersatzwerkzeuge für Werkzeugverwaltung" der Platz eines (wenn vorhandenen) Schwesterwerkzeugs als Auftrag von der Werkzeugverwaltung ausgegeben.

Kanal-MD52270: $MCS_TM_FUNCTION_MASK

Einstellungen:

SD52270: $MCS_TM_FUNCTION_MASK	
	Funktionsmaske Werkzeugverwaltung
Bit 0:	Werkzeug anlegen auf Magazinplatz nicht zugelassen. Werkzeuge können nur außerhalb des Magazins angelegt werden.
Bit 1:	Be-/Entladesperre, wenn die Maschine nicht im Reset-Zustand ist. Werkzeuge können nur be-/entladen werden, wenn der entsprechende Kanal im Reset-Zustand ist.
Bit 2:	Be-/Entladesperre bei NOT-HALT. Werkzeuge können nur be-/entladen werden, wenn NOT-HALT nicht aktiv ist
Bit 3:	Werkzeug in/aus Spindel be-/entladen gesperrt. Werkzeuge können nicht in die Spindel geladen oder aus der Spindel entladen werden.
Bit 4:	Beladen erfolgt direkt in die Spindel. Das Beladen von Werkzeugen erfolgt ausschließlich direkt in die Spindel.
Bit 5:	reserviert
Bit 6:	reserviert
Bit 7:	Werkzeug über die T-Nummer anlegen. Beim Werkzeug anlegen muss die T-Nummer des Werkzeugs eingegeben werden.
Bit 8:	Werkzeug umsetzen ausblenden. Die Funktion "Werkzeug umsetzen" wird in der Bedienoberfläche ausgeblendet.
Bit 9:	Magazin positionieren ausblenden. Die Funktion "Magazin positionieren" wird in der Bedienoberfläche ausgeblendet.
Bit 10:	Werkzeug reaktivieren mit Magazin positionieren. Vor dem Reaktivieren wird das Werkzeug auf die Beladestelle positioniert.

SD52270: $MCS_TM_FUNCTION_MASK	
Bit 11:	Werkzeug reaktivieren in allen Überwachungsarten. Beim Reaktivieren eines Werkzeugs werden alle in der NC freigegebenen Überwachungsarten zu diesem Werkzeug reaktiviert. D.h. auch die Überwachungsarten, die für das jeweilige Werkzeug nicht eingestellt sind, sondern nur im Hintergrund liegen.
Bit 12:	Werkzeug reaktivieren ausblenden. Die Funktion "Werkzeug reaktivieren" wird in der Bedienoberfläche ausgeblendet.

Voreinstellung für Technologie Drehen

Für die Technologie Drehen bestimmen zwei Maschinendaten die Funktion:

- MD22550: $MC_TOOL_CHANGE_MODE = 0

  Einstellung für ein Revolvermagazin: das neue Werkzeug wird mit der T-Funktion sofort eingewechselt. Es wird kein zusätzlicher M-Befehl verwendet. Es wird nicht zwischen Werkzeugwechsel vorbereiten und Werkzeugwechsel durchführen unterschieden.

  Die Funktion "Handwerkzeuge" ist für diesen Fall nicht freigegeben.

- MD20310: $MC_TOOL_MANAGEMENT_MASK = 81400F (Bit 16=1)

  Mit Bit 16 wird die Programmierart des Werkzeugs eingestellt:

  T = "x" mit x als Werkzeug-Bezeichner

  Tx, mit x als Platznummer des Magazins, mit dessen Werkzeug gearbeitet wird.

  Bei aktivierter Funktion wird mit T1 anstelle des Werkzeugs mit dem Bezeichner "1" das Werkzeug auf dem Platznummer 1 angewählt. Dann wird der Bezeichner des Werkzeugs auf diesem Platz ermittelt (z. B. "SCHLICHTER"). Es wird so verfahren als wäre T="SCHLICHTER" programmiert worden.

  Bei T = Platznummer ist es zulässig, dass sich kein Werkzeug auf diesem Platz befindet.

- MD10717: $MN_T_NO_FCT_CYCLE_NAME = TCHANGE

  Name des Werkzeugwechselzyklus für T-Funktions-Ersetzung.

  Eine Beschreibung finden Sie im Kapitel Beispiel: Werkzeugwechselzyklus für Drehmaschine (Seite 366)

- ISO-Dialekte

  Voreinstellung für die Programmierung der Werkzeugkorrektur:

MD10888	$MN_EXTERN_DIGITS_TOOL_NO = 0
MD10889	$MN_EXTERN_DIGITS_OFFSET_NO = 2
MD10890	$MN_EXTERN_TOOLPROG_MODE = 4

Eine Programmierung von T101 und T0101 führt zum gleichen Ergebnis: T1 H01

Magazin konfigurieren

Die Magazinkonfiguration kann wahlweise mit dem Startup Tool oder mit einem Konfigurationsprogramm erstellt werden. Das Konfigurationsprogramm wird als normales Teileprogramm angewählt und gestartet.

Nach einer Änderung der Magazinkonfiguration ist ein NC POWER ON notwendig. Die geänderte Konfiguration wird erst nach einem Neustart der NC angezeigt.

Siehe auch

Beispiele:

- Konfiguration eines Kettenmagazins mit einem Doppelgreifer (Seite 371)
- Konfiguration eines Revolvermagazins (Seite 360)

  Das Programm ist auf der Toolbox CD enthalten.

Literatur

Weitere Informationen:

- SINUMERIK 828D Listenhandbuch: Maschinendaten und Nahtstellensignale
- SINUMERIK 802D sl/828D/840D sl: Funktionshandbuch: ISO-Dialekte
  - Für ISO-Mode (G291) gilt die Beschreibung "Substitution durch einen Ersetzungszyklus bei einem Werkzeuganwahlsatz".
  - Für die Programmierung der Werkzeugkorrektur gilt die Beschreibung "Werkzeugwechsel und Werkzeugkorrekturen".

10.3.2 Abhängigkeit zwischen MD20360 und SD54215

Einstellungen für die Werkzeugverwaltung

Die Einstellungen für die Werkzeugverwaltung werden in folgendem Setting-Datum vorgenommen:

SD54215: $SNS_TM_FUNCTION_MASK_SET	
	Funktionsmaske Werkzeugverwaltung
Bit 0:	Durchmesseranzeige für rotierende Werkzeuge: Die Schneidenparameter 6 (Schneidenradius) und 15 (Verschleißradius) werden für folgende Werkzeugtypen nicht als Radiuswert, sondern als Durchmesserwert angezeigt: Typ 100 bis 299, 580, 710, 711, 712, 713, 714 und 725. Der Schneidenparameter 7 (Außenradius) wird für folgende Werkzeugtypen nicht als Radiuswert, sondern als Durchmesserwert angezeigt: Typ 140 und 714.
Bit 1:	Standarddrehrichtung für alle Drehwerkzeuge ist M4. Beim Anlegen von Drehwerkzeugen wird die Drehrichtung mit M4 vorbelegt.
Bit 2:	Beim Anlegen eines Werkzeugs erfolgt kein Namensvorschlag.

SD54215: $SNS_TM_FUNCTION_MASK_SET	
Bit 3:	Eingabesperre Werkzeugname und Werkzeugtyp bei beladenen Werkzeugen. Bei beladenen Werkzeugen können der Werkzeugname und der Werkzeugtyp nicht mehr geändert werden.
Bit 4:	Eingabesperre für beladene Werkzeuge, wenn der Kanal nicht im Reset ist.
Bit 5:	Werkzeugverschleißeingaben additiv verrechnen: Die Eingabe von Verschleißdaten erfolgt additiv zum bereits bestehenden Verschleißwert.
Bit 6:	Im Feld "Werkzeugbezeichner" werden nur numerische Eingaben zugelassen.
Bit 7:	Werkzeugüberwachungsparameter ausblenden. Die Parameter der Werkzeugüberwachung werden in der Bedienoberfläche ausgeblendet.
Bit 8:	Durchmesseranzeige für Planachse - Geometrie. Der Geometriewert der Planachse wird als Durchmesserwert angezeigt, wenn im MD20100 $DIAMETER_AX_DEF eine Planachse definiert ist und das SD42940 $TOOL_LENGTH_CONST auf 18 und das SD42950 $TOOL_LENGTH_TYPE auf 2 eingestellt sind.
Bit 9:	Durchmesseranzeige für Planachse - Verschleiß. Der Verschleißwert der Planachse wird als Durchmesserwert angezeigt, wenn im MD20100 $DIAMETER_AX_DEF eine Planachse definiert ist und das SD42940 $TOOL_LENGTH_CONST auf 18 und das SD42950 $TOOL_LENGTH_TYPE auf 2 eingestellt sind.
Bit 10:	Werkzeug beladen/umsetzen auf Zwischenspeicherplätze freischalten. Im Beladendialog kann die Magazinnummer eingegeben werden. Über die Magazinnummer 9998 kann damit auf den Zwischenspeicher zugegriffen werden.
Bit 11:	Anlegen neuer Werkzeuge auf den Greiferplätzen ist gesperrt.
Bit 12:	Messwerkzeuge werden bei der Ausführung der Funktion "Alle entladen" nicht entladen.

Abhängigkeit zwischen SD54215 und MD20360

Zwischen dem Setting-Datum SD54215 $TM_FUNCTION_MASK_SET und dem Maschinendatum MD20360 $TOOL_PARAMETER_DEF_MASK bestehen die folgenden Abhängigkeiten:

<table>
<tr><th colspan="3">Abhängigkeit</th><th rowspan="2">Beschreibung</th><th rowspan="2">Empfehlung</th></tr>
<tr><th>SD54215</th><th>⇔</th><th>MD20360</th></tr>
<tr><td>Bit 9</td><td>⇔</td><td>Bit 0</td><td>Wenn im MD20360 Bit 0 gesetzt ist, dann hat SD54215 Bit 9 keine Wirkung, da die NC den Längenverschleißwert der Planachse bereits als Durchmesserwert liefert.</td><td rowspan="2">Es wird empfohlen die Einstellung MD20360 Bit 0 und Bit 1 nicht zu verwenden.
Verwenden Sie stattdessen SD54215 Bit 9 und Bit 8.
MD20360 Bit 0 und Bit 1 wirken nur für Dreh- und Schleifwerkzeuge. Damit wird diese Einstellung auf einer Dreh-/Fräsmaschine unbrauchbar, da die Längenkomponente der Planachse für Drehwerkzeuge im Durchmesser und für Fräswerkzeuge im Radius wirkt. Bei der Verwendung des Setting-Datums geht nur die Anzeige auf Durchmesser, der interne Wert in der NC bleibt auf Radius.</td></tr>
<tr><td>Bit 8</td><td>⇔</td><td>Bit 1</td><td>Wenn im MD20360 Bit 1 gesetzt ist, dann hat SD54215 Bit 8 keine Wirkung, da die NC den Längengeometriewert der Planachse bereits als Durchmesserwert liefert.</td></tr>
<tr><td>Bit 0</td><td>⇔</td><td>Bit 11</td><td>Wenn im MD20360 Bit 11 gesetzt ist, dann hat SD54215 Bit 0 in Bezug auf den Schneidenparameter 6 (Schneidenradius) keine Wirkung, da die NC den Schneidenradius bereits als Durchmesserwert liefert.</td><td rowspan="2">Es wird empfohlen die Einstellung MD20360 Bit 11 und Bit 12 nicht zu verwenden.
Setzen Sie stattdessen SD54215 Bit 0.
MD20360 Bit 11 und Bit 12 wirken auf alle Werkzeugtypen, also auch auf Drehwerkzeuge. Damit wird diese Einstellung auf Dreh-/Fräsmaschinen unbrauchbar, da der Schneidenradius eines Drehwerkzeugs nie im Durchmesser angegeben werden sollte. Bei der Verwendung des Setting-Datums geht nur die Anzeige auf Durchmesser, der interne Wert in der NC bleibt auf Radius.</td></tr>
<tr><td>Bit 0</td><td>⇔</td><td>Bit 12</td><td>Wenn im MD20360 Bit 12 gesetzt ist, dann hat SD54215 Bit 0 in Bezug auf den Schneidenparameter 15 (Verschleißradius) keine Wirkung, da die NC den Verschleißradius bereits als Durchmesserwert liefert.</td></tr>
</table>

10.3.3 Werkzeugmessen in JOG

Werkzeugaufruf bei Werkzeugmessen in JOG

Für spezielle Anwendungen (z. B. Messzyklen in JOG) ist es notwendig, ein bestimmtes Werkzeug (z. B. Fraeser_15, Duplo-Nummer 2), unabhängig von seinem Status auf die Spindel oder den Tool Holder einzuwechseln, wenn zum Beispiel ein Werkzeug im Einsatz war und von der Werkzeugüberwachung gesperrt wurde. Jetzt soll es vor der erneuten Freigabe ausgemessen werden.

Für den Werkzeugaufruf unabhängig vom Status und mit Auswahl von Duplo- und Tool Holder-Nummer steht der NC-Befehl TCA zur Verfügung und wird bei Werkzeugmessen in JOG benutzt.

```
TCA("Werkzeugname", Duplonr., Tool Holder-Nr.)
```

TCA verhält sich bezüglich der Alarm- und Kommandoausgabe an die PLC analog zum T–Befehl.

Randbedingungen:

Bei aktiver T-Funktionsersetzung (Voreinstellung bei Technologie Drehen) sind folgende Randbedingungen zu beachten:

- TCA kann nicht substituiert werden (T-Ersetzungszyklus).
- Der in MD10717: $MN_T_NO_FCT_CYCLE_NAME definierte und bei T-Aufruf durchlaufene Zyklus wird bei dem Befehl TCA nicht gestartet.

  Um sicherzustellen, dass bei der Programmierung von TCA auch ein Werkzeugwechselzyklus startet, ist der Sprachbefehl TCA in _TCA umdefiniert (Voreinstellung NC).
- Die Maschinenfunktionen des TCHANGE.SPF müssen in den TCA.SPF Zyklus übernommen werden. Der Zyklus TCA.SPF ist auf der Toolbox CD.

Beispiel:

Das folgende Beispiel beschreibt den Ablauf:

```
PROC TCA(STRING[64] _TOOL_NAME,INT _DUPLO,INT _TH_NO)
                                          ;VERSION: 01.00.07. Oct 22, 2009
                                          ;CHANGE: 01.00.07. Oct 22, 2009
TCA(_TOOL_NAME,_DUPLO,_TH_NO)             ;Werkzeugaufruf an NC
; fügen Sie hier die Maschinenfunktionen für den Wechsel ein
M17
```

Hinweis

Die Korrekturanwahl entsprechend MD20270: $MC_CUTTING_EDGE_DEFAULT wirkt analog zum T-Befehl. TCA und D dürfen nicht in einem Satz programmiert werden.

Messen in JOG ohne elektronischen Messtaster

Für manuelle Messfunktionen in der Betriebsart JOG, für die eine drehende Spindel benötigt wird, gilt:

- Zum Zeitpunkt der Messwertübernahme muss der Kanalzustand Reset sein.
- Die Spindel kann wahlweise über das T,S,M-Menü oder mit den Bits in DB380x.DBB5006 bewegt oder positioniert werden.
- Für die Verwendung der Technologiefunktionen DB380x.DBB5006 mit fest eingestellter Drehzahl gelten folgende Voraussetzungen:

Einstellung Maschinendaten		
MD10709 $MN_PROG_SD_POWERON_INIT_TAB [0] = 0		zu initialisierende Setting-Daten
MD35035 $MA_SPIND_FUNCTION_MASK	Bit 4=0 Bit 5=1	Spindelfunktionen
SD43200 $SA_SPIND_S		Einstellwert für Drehzahl

Hinweis

Der Eintrag in SD43200 $SA_SPIND_S bleibt bei Power On erhalten.

Mit diesen Einstellungen wirkt die Spindelgeschwindigkeit für manuellen Betrieb, die in SD41200 $SN_JOG_SPIND_SET_VELO eingestellt ist, nicht mehr.

10.4 PLC Programmbausteine

10.4.1 Quittungsprozess

Informationen an die Werkzeugverwaltung

Die Werkzeugverwaltung erwartet Quittungen auf ihre Aufträge, um die realen Werkzeugpositionen stets zu verfolgen und mitzuführen. Pro Auftrag ist mindestens eine Quittung erforderlich. Für viele Anwendungen ist dies ausreichend.

Die Quittung erfolgt entweder über die in DB9902 definierte Tabelle oder nach vollständig abgeschlossenem Auftrag der Werkzeugverwaltung in einem Schritt mit der Total-Quittung (DB40xx/42xx DBX0.0) durch eine 0/1-Flanke (setzen) des entsprechenden Bits in der Anwender-Nahtstelle.

Solange das Quittungssignal ansteht, dürfen keine Änderungen an den Daten dieser Schnittstelle vorgenommen werden! Nach der Übertragung der Quittung an die Werkzeugverwaltung wird dieses Signal von der PLC-Firmware zurückgesetzt. Das Rücksetzen kann unter Umständen nach mehreren PLC-Takten erfolgen.

Wird die Werkzeugverwaltung auch über Zwischenpositionen informiert, ergeben sich weitere Vorteile:

- Information über Zwischenpositionen:

  Sind der Werkzeugverwaltung alle Zwischenpositionen von Werkzeugen bekannt, kann die Belegung des Zwischenspeichermagazins auch rückgefragt werden. Dies erleichtert die Hochlaufstrategie nach dem Aus- und Einschalten oder nach einem Kommando-Abbruch (z. B. durch Reset). Wird ein Werkzeug, das gerade ausgewechselt wird, sofort wieder benötigt, kann es ohne erst im Magazin abgelegt zu werden schon von einem Zwischenspeicherplatz wieder in die Spindel eingewechselt werden.

- Information über Magazinpositionen:

  Ist der Werkzeugverwaltung bekannt, welcher Magazinplatz sich an einer Übergabestelle (Wechselstelle zur Spindel, Beladestelle) befindet, kann sie für die Leerplatzsuche oder bei der Wahl des neuen Werkzeugs die kürzesten Wege im Magazin ermitteln. Bei Aufträgen kann die Werkzeugverwaltung die Magazinposition meist aus Zwischenquittungen (z. B. Werkzeug-Transfer zwischen realem Magazin und Zwischenspeicher) oder der Ende-Quittung (z. B. Auftrag "Magazin positionieren" abgeschlossen) erkennen. Erfolgt die Positionierung des Magazins durch das PLC-Anwenderprogramm selbst (z. B. durch HMI oder Maschinentasten) ohne Auftrag von der Werkzeugverwaltung, muss dies der Werkzeugverwaltung über asynchrone Meldungen mitgeteilt werden.

10.4.2 Quittungsarten

Werkzeug- und Magazinbewegungen

Die Werkzeugverwaltung unterscheidet zwischen synchroner Quittung und asynchroner auftragsunabhängiger Mitteilung.

Synchrone Quittung

- Quittung von Zwischenschritten eines Auftrags (Werkzeugverwaltung registriert aktuelle Positionsänderungen von Werkzeugen, Teileprogramm muss noch warten)

  Mit einer Zwischenquittung wird die Werkzeugverwaltung über Zwischenschritte zu einem Auftrag informiert. Bei Zwischenquittungen interessiert nur die Zielposition des Zwischenschrittes. Die Quellposition ist aus dem Auftrag oder der letzten Zwischenquittung bekannt. Beim Werkzeugwechsel können auch zwei Werkzeuge (neues und altes) gleichzeitig quittiert werden. Zwischenquittungen sind nur **vor** der Ende-Quittung möglich.
- Ende-Quittung zu einem Auftrag (Teileprogramm kann fortgesetzt werden)

  Zu jedem Auftrag ist eine Ende-Quittung erforderlich. Die Ende-Quittung erlaubt die Fortsetzung des Teileprogramms und gibt die Auftragschnittstelle für neue Aufträge frei. Sie sollte so früh wie möglich erfolgen (z. B. sobald sich das neue Werkzeug in der Spindel befindet und keine Kollision mehr zu erwarten ist). Weitere Schritte nach der Ende-Quittung zu einem Auftrag (z. B. den Rückweg des alten Werkzeugs ins Magazin) können der Werkzeugverwaltung asynchron mitgeteilt werden.

Asynchrone auftragsunabhängige Mitteilung

Positionsänderung eines Werkzeugs oder Magazins ("asynchrone Mitteilung", z. B. wenn die PLC eine Werkzeugposition durch Maschinensteuertafel-Bedienung ohne Werkzeugwechselauftrag verändert.)

Mit einer asynchronen Mitteilung kann die Werkzeugverwaltung unabhängig von einem Auftrag über eine Werkzeug- oder Magazinbewegung informiert werden. Eine asynchrone Mitteilung muss immer Quellposition (von) **und** Zielposition (nach) enthalten.

Werkzeugbewegungen innerhalb eines Magazins (Werkzeug umsetzen) können nur mit tatsächlich belegten Magazinplätzen durchgeführt werden. Leertransfers sind nicht zulässig. In einer Mitteilung können zwei asynchrone Transfers realisiert werden. In diesem Fall muss die Nahtstelle für Werkzeugwechsel DB42xx benutzt werden.

Wirkung der Quittungen

Wirkung der Quittungen auf den Auftrag und das Teileprogramm:

- Zwischenquittung und Ende-Quittung erfolgen synchron zum Auftrag.
  - Das Teileprogramm muss warten.
  - Es kann noch kein neuer Auftrag kommen.
- Mitteilung eines asynchronen Transfers:
  - Das Teileprogramm läuft weiter.
  - Die Mitteilung ist völlig unabhängig von irgendeinem Auftrag.

10.4.3 Quittungszustände

Quittungszustände und ihre Bedeutung:

Der Status, der die jeweilige Quittungsart kennzeichnet, ist in der nachfolgenden Tabelle angegeben:

Quittung		Bedeutung
Synchrone Ende-Quittung	1	Auftrag an angegebener Position beendet: Die Werkzeuge befinden sich auf den angegebenen Positionen. Das Teileprogramm kann fortgesetzt werden.
	3	Auftrag abgebrochen: Der Auftrag wird abgebrochen, bis dahin quittierte Positionsänderungen von Werkzeugen werden beibehalten. Das Abbruch-Kommando selbst löst keine Positionsquittungen oder -änderungen in der Werkzeugverwaltung aus.
	6	Ende-Quittung für "Werkzeug bewegen" aus realem Magazin nach einem Zwischenspeicher (Greifer, Spindel) mit Reservierung des Altplatzes im Magazin für dieses Werkzeug. Weitere Bedeutung wie Status 1.
	7	Auftrag "Werkzeugwechsel vorbereiten" wiederholen: Der Werkzeugwechsel wurde vorher eine neue Werkzeugposition mitgeteilt. Der Auftrag "Werkzeugwechsel vorbereiten" soll unter Berücksichtigung dieser Position neu berechnet werden. Nur zulässig bei noch nicht quittiertem Vorbereitungskommando.
	99	Total-Quittung: Auftrag beendet, alle Positionen erreicht. Alle beteiligten Werkzeuge befinden sich auf den im Auftrag vorgegebenen Positionen. Das Teileprogramm kann fortgesetzt werden. Alle Zielpositionen aus dem Auftrag sind erreicht.
Synchrone Zwischenquittung	105	Zwischenposition für Werkzeug: Die Werkzeuge wurden von der im Auftrag angegebenen Quellposition oder von der zuletzt quittierten Zwischenposition auf die angegebene Zielposition bewegt.
Asynchronen Transfer mitteilen	201	Werkzeugbewegung mitteilen: Das Werkzeug wurde von der Quellposition auf die angegebene Zielposition bewegt. Bei Bewegung von einem Platz im realen Magazin auf einen Zwischenspeicherplatz, wird der Quellplatz für das Werkzeug reserviert. Beachten Sie hierzu auch MD20310, Bit 24 (Seite 328).
	204	Magazinposition mitteilen: Der Magazinplatz befindet sich an der zum angegebenen Zielplatz gehörenden Wechsel-/Belade-/Entladestelle.

Übersicht über ausgewertete Tabellen-Parameter

Quittungsstatus		1	3	6	7	99	105	201	204
Neues Werkzeug	Transferschritt	x	-	x	-	-	x	x	x
	• von Magazin	-	-	-	-	-	-	xx	xx
	• von Platz	-	-	-	-	-	-	xx	xx
	• nach Magazin	xx	-	xx	-	-	xx	xx	zz
	• nach Platz	xx	-	xx	-	-	xx	xx	zz
Altes Werkzeug	Transferschritt	x	-	-	-	-	x	x	-
	• von Magazin	-	-	-	-	-	-	xx	-
	• von Platz	-	-	-	-	-	-	xx	-
	• nach Magazin	xx	-	-	-	-	xx	xx	-
	• nach Platz	xx	-	-	-	-	xx	xx	-

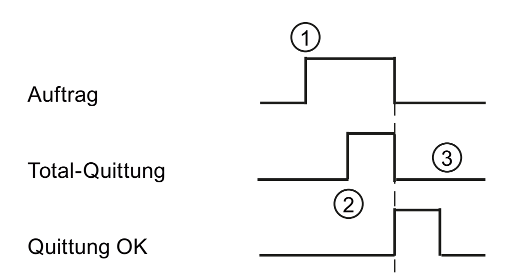
Legende:

-	Datum nicht relevant
x	Nummer (1…n) des Transferschrittes aus der Transferschritt-Tabelle
xx	Magazinnummer, Platznummer des Werkzeugs
zz	Magazinnummer, Platznummer der Belade-/Entlade- oder Wechselstelle

Die Statusübersicht gibt folgende Informationen aus:

- Die Status 1, 6, 105, 201 und 204 sind in der Quittungsschritt-Tabelle entsprechend ihrer Bedeutung zusammen mit Transferschritten zu sinnvollen Quittungsschritten zusammenzusetzen.
- Wird Status 1 mit beiden Transferschrittnummern = 0 kodiert, dient dieser Quittungsschritt als Ende-Quittung des bis zu diesem Zeitpunkt durch Zwischenquittungen erreichten Standes.
- Wurde ein Werkzeug aus einem realen Magazin in einen Zwischenspeicher bewegt (Umsetzen, MVTOOL), wird bei Quittung mit Status 6 oder Total-Quittung der Quellplatz für dieses Werkzeug reserviert ($TC_MPP4 Bit 1 und Bit 2). Das Verhalten ist gleich wie bei der Entnahme des Werkzeugs aus dem Magazin bei Werkzeugwechsel. Mit Status 1 erfolgt bei Umsetzen oder MVTOOL keine Reservierung des Quellplatzes.
- Die Status 3 und 7 brauchen in der Quittungsschritt-Tabelle nur einmal kodiert zu werden, da keine Transferschritte ausgewertet werden.
- Status 99 braucht nicht kodiert werden, er wird durch das Bit der "Total-Quittung" vorgegeben.

Typischer Ablauf eines beliebigen Auftrags mit Total-Quittung



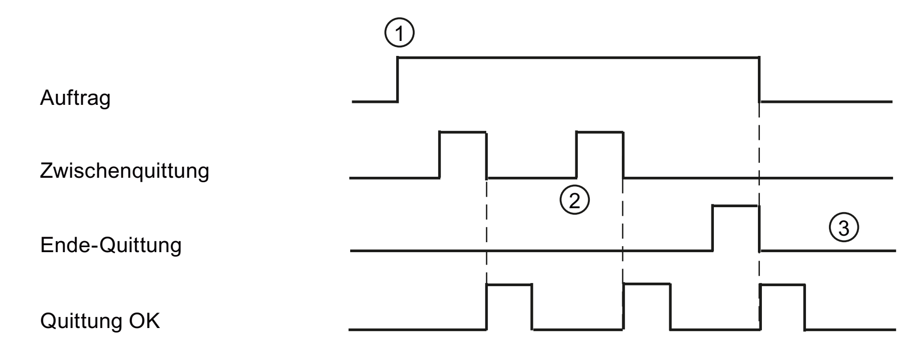
Signalbeschreibung:

① Das PLC-Anwenderprogramm erkennt an der 0/1 Flanke des Signals DB43xx.DBX0.0 (Auftrag), dass ein neuer Auftrag von der Werkzeugverwaltung übergeben wurde.

② Das PLC-Anwenderprogramm setzt das Quittungssignal in DB42xx.DBX0.0 (Total-Quittung). Mit Auslösen der 0/1 Flanke beginnt die PLC-Firmware mit der Übertragung der Quittung an die Werkzeugverwaltung.

③ Nach erfolgreicher Übertragung der Quittung an die Werkzeugverwaltung wird von der PLC-Firmware das Signal Quittung OK einen PLC Takt auf 1 gesetzt und gleichzeitig das Signal Auftrag sowie das Quittungsbit auf 0 zurückgesetzt.

Typischer Ablauf eines beliebigen Auftrags mit Zwischenquittung und Ende-Quittung



Signalbeschreibung:

① Das PLC-Anwenderprogramm erkennt an der 0/1 Flanke des Signals DB43xx.DBX0.0 (Auftrag), dass ein neuer Auftrag von der Werkzeugverwaltung übergeben wurde.

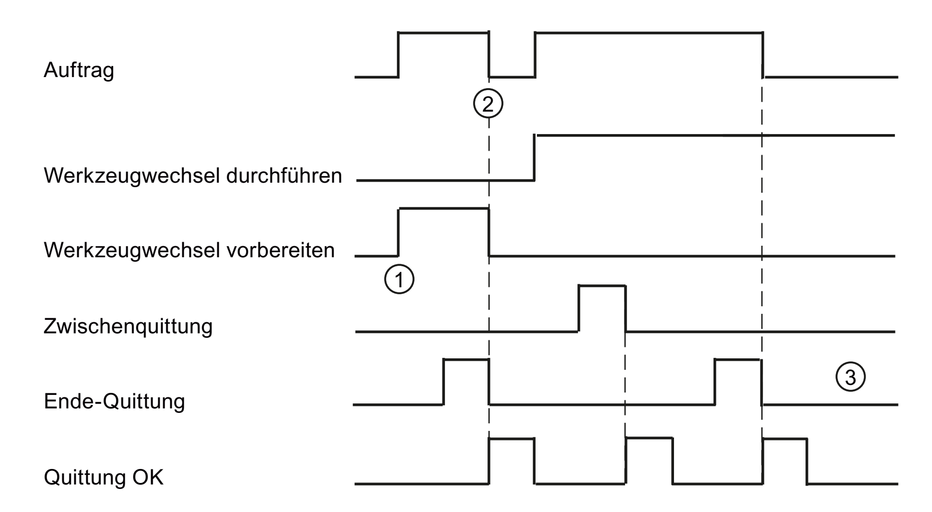
② Das PLC-Anwenderprogramm quittiert die in den DB9900, DB9901 und DB9902 projektierten Transferschritte mit Quittungsstatus 105. Die Positionen der Werkzeuge werden anhand der Transferschritte der Quittungen von der Werkzeugverwaltung aktualisiert.

③ Das PLC-Anwenderprogramm quittiert mit Quittungsstatus 1 die Ausführung des Auftrags. Nach erfolgreicher Übertragung der Quittung an die WZV wird von der PLC-Firmware das Signal Quittung OK einen PLC-Takt auf 1 gesetzt und gleichzeitig das Signal Auftrag sowie das Quittungsbit auf 0 zurückgesetzt.

Ablauf der Werkzeugverwaltung mit Satz-Splitting (Maschinendaten-Einstellung Fräsen)

```
Txx M6 ;
```

Mit M6 wird das Programm L6 aufgerufen (Grundeinstellung)



Signalbeschreibung:

① Das PLC-Anwenderprogramm bekommt einen neuen Auftrag. Der Auftrag "Werkzeugwechsel vorbereiten" und der Auftrag "Werkzeugwechsel durchführen" kommen nacheinander. Txx und M206 wurden in getrennten NC Sätzen programmiert. In der Nahtstelle in DB43xx steht nur der Auftrag "Werkzeugwechsel vorbereiten". Der Auftrag "Werkzeugwechsel durchführen" wird erst nach Ende-Quittung des Auftrags für das Vorbereiten ausgegeben.

② Mit Quittung des Auftrags "Werkzeugwechsel vorbereiten" wird das Bit DB43xx.DBX0.0 (Auftrag) zurückgesetzt. Falls der Wechselbefehl (M206) bereits den Hauptlauf der NC durchlaufen hat, wird anschließend sofort der neue Auftrag an der Nahtstelle ausgegeben.

③ Der Auftrag "Werkzeugwechsel durchführen" wird als normaler Auftrag quittiert. Es wird die Ende-Quittung mit OK zurück gemeldet und gleichzeitig das Bit für den Auftrag zurückgesetzt.

Die Beschreibung des Auftrags ("Werkzeugwechsel durchführen" und "Werkzeugwechsel vorbereiten") wird nicht zurückgesetzt. Byte 1 von DB43xx wird erst mit dem nächsten Auftrag neu geschrieben.

Hinweis

MD20270, MD20310:

Das Verhalten der Nahtstelle in DB43xx.DBB1 sowie der NC Satzverarbeitung wird mit der Einstellung in MD20270: $MC_CUTTING_EDGE_DEFAULT und MD20310: $MC_TOOL_MANAGEMENT_MASK Bit 5, 6, 7 und 8 beeinflusst.

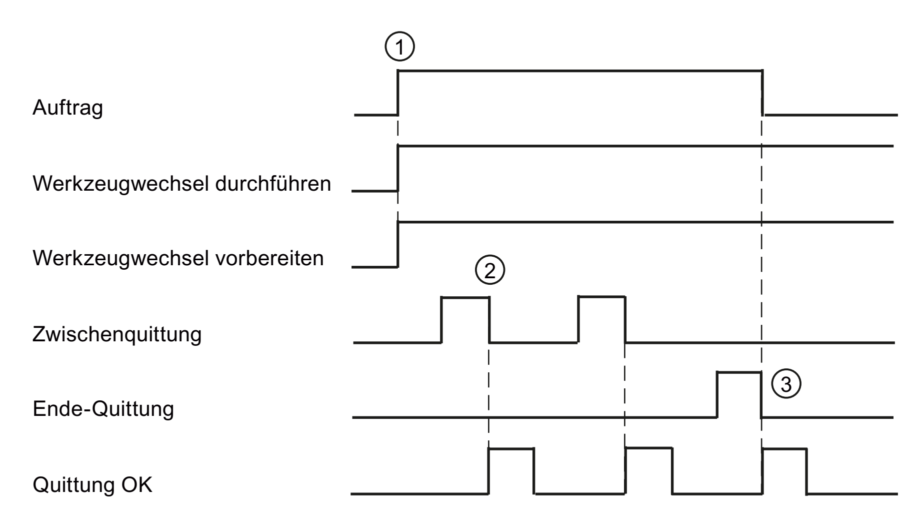
Der hier beschriebene Ablauf entspricht der Voreinstellung des Maschinendatums.

Ablauf der Werkzeugverwaltung ohne Satz-Splitting (Maschinendaten-Einstellung Fräsen)

```
Txx Myy ;
```

Myy ist die Einstellung aus MD22560: $MC_TOOL_CHANGE_M_CODE

Diese Art der Programmierung wird nicht empfohlen!



Signalbeschreibung:

① Das PLC-Anwenderprogramm bekommt einen neuen Auftrag. Der Auftrag "Werkzeugwechsel vorbereiten" und der Auftrag "Werkzeugwechsel durchführen" kommen gemeinsam. Txx und M206 wurden in einem NC Satz programmiert.

② Es werden mehrere Zwischenschritte quittiert. Der Zustand des Auftrags bleibt unverändert. Die Positionen der Werkzeuge werden anhand der Transferschritte der Quittungen von der Werkzeugverwaltung aktualisiert.

③ Es wird die Ende-Quittung mit OK zurückgemeldet und gleichzeitig das Bit für den Auftrag zurückgesetzt.

Die Beschreibung des Auftrags ("Werkzeugwechsel durchführen" und "Werkzeugwechsel vorbereiten") wird nicht zurückgesetzt. Byte 1 von DB43xx wird erst mit dem nächsten Auftrag neu geschrieben.

Hinweis

MD20310: $MC_TOOL_MANAGEMENT_MASK

Das Verhalten der Nahtstelle in DB43xx.DBB1 wird mit der Einstellung in MD20310 Bit 10 beeinflusst.

Der hier beschriebene Ablauf entspricht der Voreinstellung des Maschinendatums.

10.4.4 Schritt-Tabellen projektieren

Schritt-Tabellen projektieren

Die Transferschritt-Tabellen (TM_CTS, DB9900 und TM_VTS, DB9901) und die Quittungsschritt-Tabelle (TM_ACK, DB9902) finden Sie im Programming Tool unter "Bibliotheken" → "Spezielle Datenbausteine". Mit einem Doppelklick werden die Bausteine in das Projekt kopiert.

Die Struktur der Datenbausteine ist fest vorgegeben.

Die Bausteine sind noch nicht mit den notwendigen Daten gefüllt und müssen vom Anwender im Programming Tool über den Menübefehl "Ansicht" → "Datenbaustein" bearbeitet werden. Die Projektierung der konstanten Tabellen (TM_CTS, DB9900 und TM_ACK, DB9902 erfolgt durch Schreiben der Anfangswerte der Datenbausteine im Programming Tool.

Gemeinsam mit dem PLC-Anwenderprogramm werden die Anfangswerte der Datenbausteine in die Steuerung geladen. Geänderte Anfangswerte werden erst nach einem Neustart der PLC aktiv.

Projektierung der Transferschritte

Änderungen von Werkzeug- und Magazinpositionen durch das PLC-Anwenderprogramm müssen der Werkzeugverwaltung mitgeteilt werden. Als Hilfsmittel dient dazu eine Tabelle aller mechanischen Einzelbewegungen, die quittiert/mitgeteilt werden sollen. Die Tabelle enthält für jeden Werkzeug-Transfer jeweils die Start- und Zielposition des Werkzeugs oder für die Positionierung eines Magazinplatzes an eine Übergabestelle (Wechsel-, Be-, Entladestelle) die Magazinposition und die Bezeichnung der Übergabestelle.

- Die Transferschritte 1 … 64 sind fest in TM_CTS (DB9900) projektiert und nur durch Neuladen änderbar.
- Die Transferschritte 101 … 164 in TM_VTS (DB9901) können vom PLC-Anwenderprogramm ganz oder teilweise überschrieben werden (z. B. durch das Eintragen des aktuellen Magazinplatzes).

Codierung für Position aus Auftrag

Die Plätze realer Magazine werden in der konstanten Transferschritttabelle nicht mit ihren tatsächlichen Werten (z. B. 1/14 für Magazin 1 Platz 14) sondern mit symbolischen Werten (0/1) oder (0/2) bezeichnet. Andernfalls würde die Transferschritttabelle bei großen Magazinen riesige Ausmaße annehmen.

Diese symbolischen Werte haben folgende Bedeutung:

Magazin / Platz	Bedeutung
(0 / 1)	Die Quellposition des einzigen oder neuen Werkzeugs aus dem Auftrag soll benutzt werden.
(0 / 2)	Die Zielposition des alten Werkzeugs aus dem Auftrag soll benutzt werden.
(0 / 3)	Die Zielposition des einzigen oder neuen Werkzeugs aus dem Auftrag soll benutzt werden.

Diese symbolische Notationsform kann nur für synchrone Zwischenquittungen und Ende-Quittungen benutzt werden, da asynchrone Mitteilungen keinen Auftrag als Bezug haben.

Beispiel: Konstante Transferschritt-Tabelle

Transferschritt	Adresse DB9900	Name	Anfangswert	Kommentar
1	0.0	SrcMag_1	0	Quell-Magazinnummer des Transferschritts
	2.0	SrcPos_1	1	Quell-Positionsnummer des Transferschritts
	4.0	DstMag_1	0	Ziel-Magazinnummer des Transferschritts
	6.0	DstPos_1	1	Ziel-Positionsnummer des Transferschritts
2	8.0	SrcMag_2	0	Quell-Magazinnummer des Transferschritts
	10.0	SrcPos_2	1	Quell-Positionsnummer des Transferschritts
	12.0	DstMag_2	9998	Ziel-Magazinnummer des Transferschritts
	14.0	DstPos_2	2	Ziel-Positionsnummer des Transferschritts

Beispiel als komplette Schritte

Transfer-schritt	von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
1	0	1	0	1	Werkzeug vorbereiten: Magazin wird zur Wechselstelle des neuen Werkzeugs positioniert.
2	0	1	9998	2	Werkzeugwechsel: Werkzeug von Magazin nach Greifer 1
3	9998	1	9998	3	Werkzeugwechsel: Werkzeug von Spindel nach Greifer 2
4	9998	2	9998	1	Werkzeugwechsel: Werkzeug von Greifer 1 nach Spindel
5	9998	3	0	2	Werkzeugwechsel: Werkzeug von Greifer 2 nach Magazin

Beispiel: Variable Transferschritt Tabelle

Transfer-schritt	Adresse DB9901	Name	Anfangswert	Kommentar
101	0.0	SrcMag_101	1	Quell-Magazinnummer des Transferschritts
	2.0	SrcPos_101	0	Quell-Positionsnummer des Transferschritts
	4.0	DstMag_101	9998	Ziel-Magazinnummer des Transferschritts
	6.0	DstPos_101	1	Ziel-Positionsnummer des Transferschritts
102	8.0	SrcMag_102	1	Quell-Magazinnummer des Transferschritts
	10.0	SrcPos_102	0	Quell-Positionsnummer des Transferschritts
	12.0	DstMag_102	9998	Ziel-Magazinnummer des Transferschritts
	14.0	DstPos_102	2	Ziel-Positionsnummer des Transferschritts

Beispiel als komplette Schritte

Transfer-schritt	von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
101	1	0	9998	1	"Werkzeugwechsel vorbereiten": Magazinplatz wird zur Wechselstelle positioniert. Die Quellposition muss vom PLC-Anwenderprogramm eingetragen werden.
102	1	0	9998	2	"Werkzeugwechsel vorbereiten": Werkzeug von Magazin nach Zwischenspeicher. Die Quellposition muss vom PLC-Anwenderprogramm eingetragen werden.

10.4.5 Quittungsschritte projektieren

Projektierung der Quittungsschritte

Zum Quittieren von Werkzeug und Magazinbewegungen stellt die PLC 31 Quittungsschritte bereit, die über entsprechende Bits auf der Anwendernahtstelle zu aktivieren sind. Die Daten dieser Quittungsschritte (mit Ausnahme des Sonderfalls Quittungsschritt 9: Total-Quittung) werden in der Quittungsschritt-Tabelle TM_ACK (DB9902) hinterlegt. Zu einem Quittungsschritt werden die Transferschritte (Nummer des Transferschritts aus der Transferschritt-Tabelle) für altes und neues Werkzeug und ein Quittungsstatus zusammengefasst.

Wichtig ist es, in diese Tabelle einen Quittungsschritt mit dem Status 3 aufzunehmen, damit eventuelle Fehler zurückgesetzt werden können. Die Nummern für den Transfer in diesem Quittungsschritt sind 0.

Sonderbedeutung für Transferschritt 0

Es werden nur die dem Quittungsschritt zugeordneten Transferschritte ausgeführt. Ist nur ein oder kein Transferschritt zugeordnet wird für das Werkzeug mit Transferschritt = 0 kein Werkzeugtransfer ausgeführt. Das Werkzeug ist nicht vorhanden oder verbleibt auf seinem bisherigen Platz.

Beispiel: Quittungsschritt-Tabelle

Quittungs-schritt	Adresse DB9902	Name	Anfangs-wert	Kommentar
1	0.0	TsNewT_1	0	Transferschritt-Nummer des neuen Werkzeugs
	1.0	TsOldT_1	0	Transferschritt-Nummer des alten Werkzeugs
	2.0	State_1	3	Status an NCK
2	4.0	TsNewT_2	1	Transferschritt-Nummer des neuen Werkzeugs
	5.0	TsOldT_2	0	Transferschritt-Nummer des alten Werkzeugs
	6.0	State_2	1	Status an NCK

Beispiel als komplette Schritte

Quittungs-schritt	Transferschritt		Quittungs-status	Bemerkung
	neues Werkzeug	altes Werkzeug		
1	0	0	3	Kommando "Auftrag abbrechen"
2	2	0	1	Werkzeug vorbereiten: Werkzeug von Magazin (Wechselstelle) in den Greifer 1 übernommen.

Die Quittung erfolgt durch Setzen des entsprechenden Bits in der Anwendernahtstelle:

- DB40xx für Beladen/Entladen, Umsetzen oder Magazin positionieren
- DB42xx für "Werkzeugwechsel vorbereiten" und "Werkzeugwechsel durchführen"

Das Quittungsbit wird von der PLC-Firmware einen PLC-Zyklus nach der Verarbeitung zurückgesetzt.

Im gleichen Datenbaustein, in dem quittiert wurde, wird für einen PLC Zyklus die Rückmeldung in Bit 100.0 (Quittung OK) oder als statisches Signal Bit 100.1 (Quittung fehlerhaft), bei fehlerhafter Quittung in Byte 104 der Fehlerstatus und in den Bytes DBB108 bis DBB111 die zuletzt gesetzten Quittierbits ausgegeben. Anhand dieser Bits kann der Anwender erkennen, welcher Quittungsschritt den Fehler ausgelöst hat. Wurden unrichtigerweise vom PLC-Anwenderprogramm mehrere Quittungsbits gesetzt, werden diese auch 1:1 in das Abbild eingetragen. Das Rücksetzen des Fehlerstatus erfolgt mit dem Quittieren des Fehlers durch den Anwender in DB40xx.DBX9.0 oder DB42xx.DBX9.0.

10.4.6 PLC-Anwenderprogramm anpassen

PLC-Anwenderprogramm anpassen

Die Steuerung der mechanischen Vorgänge, Überwachung und Verhinderung von möglichen Kollisionen und die Quittung der Positionsänderungen der Werkzeuge sind Aufgabe des PLC-Anwenderprogramms.

Siehe auch

Im PLC Projekt auf der Toolbox CD finden Sie Beispiele für Transferschritte und deren Quittung für eine Drehmaschine mit Revolvermagazin und eine Fräsmaschine mit Kettenmagazin und Doppelgreifer:

- Anwender-Beispiel für Fräsmaschine (Seite 371)
- Anwender-Beispiel für Drehmaschine (Seite 360)

Diese Bausteine sind Funktionsbeispiele für die Quittung verschiedener Aufträge der Werkzeugverwaltung.

Aufträge quittieren

Viele Werkzeugbewegungen oder Aufträge der Werkzeugverwaltung können ohne vorherige Zwischenquittung direkt mit Total-Quittung an Bit 0.0 der Nahtstelle quittiert werden.

Zum Beispiel:

- Drehungen eines Revolvermagazins
- Beladen/Entladen (nur bei Systemen ohne zusätzliche Zwischenspeicher wie Handling-Systeme, Lader usw.)
- Wechsel von Handwerkzeugen
- Magazin positionieren

Regeln beim Quittieren

Sind Zwischenschritte sinnvoll, müssen einige Regeln beim Quittieren eingehalten werden:

Vom PLC-Anwenderprogramm muss sichergestellt sein, dass alle Quittungen ordnungsgemäß an die Werkzeugverwaltung übermittelt werden.

- Es darf immer nur ein Quittungssignal an die Werkzeugverwaltung gesendet werden.
- Synchrone Quittungen sind nur bei anstehendem Auftrag zulässig.
- Es dürfen nur gültige Transferschritt-Nummern verwendet werden (1 - 64, 100 - 164). Bei asynchronen Mitteilungen ist bei Status 201 mindestens ein Transferschritt, bei Status 204 ein Transferschritt für das Neuwerkzeug anzugeben.
- Codierte Positionen in den Transferschritten dürfen nur für synchrone Quittungen und nur mit den Werten 0/1, 0/2 oder 0/3 verwendet werden.
- Es dürfen keine unerlaubten Quittungsstatus verwendet werden.

- Magazin positionieren mit Auftrag darf nur eine synchrone Quittung (Ende-Quittung) erhalten. Zwischenpositionen müssen mit asynchronen Mitteilungen an die Werkzeugverwaltung gemeldet werden
- Quittungssignale müssen vom PLC-Grundprogramm zurückgesetzt werden. Nach Setzen eines Quittungsbits darf die Anwendernahtstelle bis zur Rückmeldung in DB41xx/DB43xx DBB100 nicht verändert werden!
- asynchrone Mitteilungen mit zwei Transferschritten müssen auf der Schnittstelle für Werkzeugwechsel (DB42xx) quittiert werden.

10.4.7 Auskunft zum Magazinplatz

Übersicht

Über die bestehende Nahtstelle NC-Dienste (DB1200) besteht die Möglichkeit, bis zu 8 NC-Variablen in einem Auftrag zu lesen.

Variable $TC_MPP2 mit Index 7: Platztyp

Parametrierung:

Variable aus NCK lesen	Adresse	Signal	Gültige Werte
Auftrag	DB1200.DBX0.0	Start	0/1
	DB1200.DBX0.1	Variable schreiben	0
	DB1200.DBB1	Anzahl der Variablen	1 … 8
Parameter	DB120x.DBW1000	Variablen Index	7
	DB120x.DBW1002	Platznummer	1 … 31999
	DB120x.DBW1004	Magazinnummer	1 … 9999
Ergebnis	DB1200.DBX2000.0	Auftrag beendet	0/1
	DB1200.DBX2000.1	Fehler im Auftrag	0/1
	DB120x.DBX3000.0	Variable ist gültig	0/1
	DB120x.DBB3001	Zugriffergebnis	0/3/5/10
	DB120x.DBW3004	Daten von NCK Variable	n

Legende:

n > 0:	Platztyp für virtuellen Platz
n = 0:	"match all" (Zwischenspeicher)
n = 9999:	undefiniert (kein virtueller Platz)

Variable $TC_MPP4 mit Index 8: Platzzustand

Parametrierung:

Variable aus NCK lesen	Adresse	Signal	Gültige Werte
Auftrag	DB1200.DBX0.0	Start	0/1
	DB1200.DBX0.1	Variable schreiben	0
	DB1200.DBB1	Anzahl der Variablen	1 … 8
Parameter	DB120x.DBW1000	Variablen Index	8
	DB120x.DBW1002	Platznummer	1 … 31999
	DB120x.DBW1004	Magazinnummer	1 … 9999
Ergebnis	DB1200.DBX2000.0	Auftrag beendet	0/1
	DB1200.DBX2000.1	Fehler im Auftrag	0/1
	DB120x.DBX3000.0	Variable ist gültig	0/1
	DB120x.DBB3001	Zugriffergebnis	0/3/5/10
	DB120x.DBW3004	Daten von NCK Variable	n

Legende:

n = 1	gesperrt
n = 2	frei (<> belegt)
n = 4	reserviert für Werkzeug im Zwischenspeicher
n = 8	reserviert für zu beladendes Werkzeug
n = 16	belegt im linken Halbplatz
n = 32	belegt im rechten Halbplatz
n = 64	belegt im oberen Halbplatz
n = 128	belegt im unteren Halbplatz

Variable $TC_MPP6 mit Index 9: T-Nummer des Werkzeugs auf diesem Platz

Parametrierung:

Variable aus NCK lesen	Adresse	Signal	Gültige Werte
Auftrag	DB1200.DBX0.0	Start	0/1
	DB1200.DBX0.1	Variable schreiben	0
	DB1200.DBB1	Anzahl der Variablen	1 … 8
Parameter	DB120x.DBW1000	Variablen Index	9
	DB120x.DBW1002	Platznummer	1 … 31999
	DB120x.DBW1004	Magazinnummer	1 … 9999
Ergebnis	DB1200.DBX2000.0	Auftrag beendet	0/1
	DB1200.DBX2000.1	Fehler im Auftrag	0/1

Variable aus NCK lesen	Adresse	Signal	Gültige Werte
	DB120x.DBX3000.0	Variable ist gültig	0/1
	DB120x.DBB3001	Zugriffergebnis	0/3/5/10
	DB120x.DBW3004	Daten von NCK Variable	n
	n = T-Nummer des Werkzeugs auf parametriertem Platz		

Fehlerfall (für alle Variablen)

Im Fehlerfall bleibt DB120x.DBX3000.0 = 0 und es erfolgt ein Eintrag im Zugriffsergebnis:

Werte in DB120x.DBB3001	
0	kein Fehler
3	Objektzugriff nicht erlaubt
5	ungültige Adresse
10	Objekt nicht existent

10.4.8 PI Dienst: TMMVTL

Funktion

Mit dem PI Dienst TMMVTL ist es von PLC aus möglich, einen Auftrag zum Umsetzen eines Werkzeugs anzustoßen. Nach fehlerfreiem "PI Start" führt die Werkzeugverwaltung für das Werkzeug auf dem angegebenen Quellplatz eine Leerplatzsuche im Zielmagazin durch. Anschließend erhält die PLC einen Auftrag zum Umsetzen des Werkzeugs (Anwendernahtstelle DB41xx.DBB0).

Parametrierung

Programm-Instanz Dienste im NCK-Bereich starten:

PI Dienst	Adresse	Signal	Gültige Werte
Auftrag	DB1200.DBX4000.0	Start	0/1
	DB1200.DBB4001	PI Index	5
Parameter	DB1200.DBW4004	Werkzeugnummer (interne T-Nummer)	1 ... 31999
	DB1200.DBW4006	Quell-Platznummer	1 ... 31999
	DB1200.DBW4008	Quell-Magazinnummer	1 ... 31999
	DB1200.DBW4010	Ziel-Platznummer	-1
	DB1200.DBW4012	Ziel-Magazinnummer	1 ... 32000

PI Dienst	Adresse	Signal	Gültige Werte
Ergebnis	DB1200.DBX5000.0	Auftrag beendet	0/1
	DB1200.DBX5000.1	Fehler im Auftrag	0/1

Das Werkzeug kann wahlweise über T-Nummer oder über Platz- und Magazinnummer angegeben werden. Eine nicht benutzte Spezifikation trägt den Wert -1.

Mit Parameter Ziel-Platznummer =-1 wird entsprechend der eingestellten Suchstrategie im gesamten Magazin ein Leerplatz für das Werkzeug gesucht. Ist ein Zielplatz angegeben, wird geprüft, ob der Platz mit der angegebenen Ziel-Platznummer für das Werkzeug frei und geeignet ist.

Bei Ziel-Magazinnummer = -1 wird für Werkzeuge auf einem Zwischenspeicher entsprechend der sich aus $TC_MDP2 ergebenden Zuordnung gesucht.

Anwendung

Beispiele:

- Bei Verwendung von Zwischenspeichern zum Rücktransport des Werkzeugs (zum Beispiel Toolboy und/oder Shifter) kann im Zuge des asynchronen Rücktransports eine explizite Leerplatzsuche im Magazin notwendig sein. In diesem Fall muss sich die PLC nicht den Ursprungsplatz merken, dieser PI Dienst sucht nach einem geeigneten Platz.
- Aus einem Hintergrundmagazin soll ein Werkzeug in das Vordergrundmagazin geholt werden.

10.5 Beispiel: Beladen / Entladen

Programmierung

Zum Beladen werden Werkzeuge direkt in das Magazin oder die Spindel gesteckt, zum Entladen direkt aus dem Magazin entnommen. Es reicht in der Regel eine einmalige Quittung des Bedieners oder des PLC-Anwenderprogramm (Werkzeughalter ist wieder geschlossen) als Meldung, dass der Vorgang beendet ist. Die Projektierung eines Transferschritts ist nicht notwendig. Es kann die Total-Quittung in DB40xx.DBX0.0. gesetzt werden.

Quittung an Werkzeugverwaltung:

Quittungsschritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
xxx	DB4000.DBX0.0	--	--	(99)

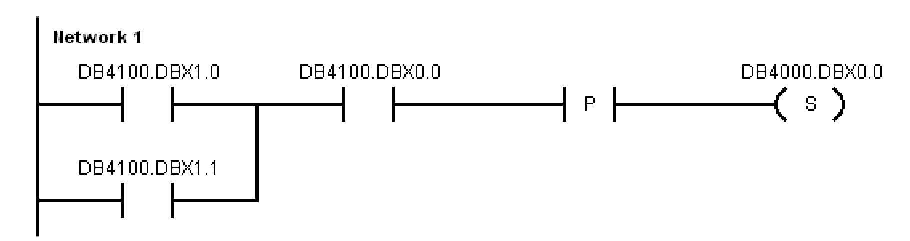


Bild 10-2 Programmierung im PLC-Anwenderprogramm

Beladevorgänge über Handling-Systeme oder der Transport eines Werkzeugs von der Spindel ins Magazin können über weitere asynchrone Mitteilungen erfolgen.

Im Weiteren sind verschiedene Abläufe eines Beladevorgangs beschrieben:

Beladen über die Spindel mit Vorwahl eines Magazinplatzes

Sie können ein neues Werkzeug direkt auf dem gewünschten freien Magazinplatz anlegen oder ein Werkzeug aus der Werkzeugliste, das sich nicht im Magazin befindet, mit dem Dialog "Beladen" auf einen Magazinplatz bringen.

1. Hier wird immer ein Auftrag der Werkzeugverwaltung an der Nahtstelle für die Beladestelle ausgegeben. Dieser Auftrag muss quittiert werden.
2. Danach wechseln Sie dieses, sich noch nicht im Magazin befindliche Werkzeug, mit Txx M6 oder über einen asynchronen Transfer in die Spindel.
3. Stecken Sie das Werkzeug danach von Hand in die Spindel und legen es mit T0 M6 in das Magazin ab.

Dieser Ablauf kann **immer** realisiert werden unabhängig davon, ob Handwerkzeuge zugelassen sind. Sind Handwerkzeuge erlaubt, muss bei Beladen über die Spindel dieser Ablauf immer eingehalten werden.

Die Werkzeuge können übergroß oder festplatz-codiert sein.

Beladen über die Spindel ohne Vorwahl eines Magazinplatzes

Sie können ein neues Werkzeug direkt auf der Spindel anlegen oder ein Werkzeug aus der Werkzeugliste, das sich nicht im Magazin befindet, mit dem Dialog "Beladen" auf die Spindel bringen.

1. Hier wird immer ein Auftrag der Werkzeugverwaltung an der Nahtstelle für die Beladestelle ausgegeben. Dieser Auftrag muss quittiert werden.
2. Stecken Sie das Werkzeug danach von Hand in die Spindel und legen es mit T0 M6 in das Magazin ab. Von der Werkzeugverwaltung wird ein freier Platz ausgewählt auf dem das Werkzeug abgelegt werden kann.

Dieser Ablauf ist nur möglich, wenn die Funktion Handwerkzeuge nicht gesetzt ist MD22562: $MC_TOOL_CHANGE_ERROR_MODE Bit 1=0 (Voreinstellung).

Die Werkzeuge können übergroß oder festplatz-codiert sein.

Beladen direkt in das Magazin

Positionieren Sie den gewünschten Magazinplatz zur Beladestelle. Sie können ein neues Werkzeug direkt auf dem gewünschten freien Magazinplatz anlegen oder ein Werkzeug aus der Werkzeugliste, das sich nicht im Magazin befindet, mit dem Dialog "Beladen" auf den gewählten Magazinplatz bringen.

1. Hier wird immer ein Auftrag der Werkzeugverwaltung an der Nahtstelle für die Beladestelle ausgegeben. Dieser Auftrag muss quittiert werden.
2. Stecken Sie jetzt das Werkzeug in das Magazin.

Für diesen Ablauf gibt es keine Einschränkungen oder Randbedingungen.

10.6 Beispiel: Handwerkzeuge wechseln

Programmierung

In MD22562: $MC_TOOL_CHANGE_ERROR_MODE Bit 1=1 ist vom NC-Teileprogramm zusätzliche Werkzeuge ohne Magazinzuordnung ausgewählt worden. Das ausgewählte Werkzeug muss von Hand in die Maschine eingesetzt und nach der Bearbeitung wieder von Hand entnommen werden ("Handwerkzeuge").

Der Bediener hat dafür zu sorgen, dass der Datensatz des Werkzeugs, das er auf die Spindel steckt, sich im NCK befindet, oder dass er zu dem im NCK abgelegten Datensatz das passende Werkzeug auf die Spindel bringt.

Hinweis

Für die Einhaltung der Sicherheitsvorschriften muss der Anwender über das PLC-Anwenderprogramm sorgen.

Dem PLC-Anwenderprogramm wird bei einem Auftrag zu einem Werkzeugwechsel mit DB43xx.DBX1.5 und DBX1.6 mitgeteilt, ob ein Handwerkzeug beteiligt ist. Der Bediener erhält mit Alarm 17212: "Kanal %1, Handwerkzeug %2, Duplo-Nr. %3, einwechseln auf Werkzeughalter %4". oder Alarm 17214: "Handwerkzeug von Spindel/Werkzeughalter entnehmen" die Aufforderung den Werkzeugwechsel durchzuführen.

Die Alarme werden durch die Quittung der PLC nach Werkzeugwechsel zurückgesetzt.

Ausgangsposition 1

Handwerkzeug in der Spindel soll gegen anderes Handwerkzeug gewechselt werden

Auftrag von Werkzeugverwaltung an PLC-Anwenderprogramm (Werkzeugwechsel):

DB4300.DBX0.0, DBX1.2, DBX1.5 und DBX1.6 ("Werkzeugwechsel vorbereiten")

DB43xx.DBW6	Quell-Magazinnummer	9999
DB43xx.DBW8	Quell-Platznummer	1
DB43xx.DBW10	Ziel-Magazinnummer	9999
DB43xx.DBW12	Ziel-Platznummer	1

Quittung "Werkzeugwechsel vorbereiten":

Quittungsschritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
xxx	DB4200.DBXx.x	0	0	1

Damit steht auf der Nahtstelle:

DB4300.DBX0.0 /1.1, DBX1.5 und DBX1.6 (durchführen")

Der Auftrag bleibt sonst unverändert, die Werkzeuge sind noch auf den Ausgangspositionen.

Zur Quittung der Entnahme des alten Werkzeugs kann ein Zwischenschritt eingebaut werden.

Synchrone Quittung: das Werkzeug befindet sich nicht mehr in der Spindel.

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
6 DB9900.DBW40	9998	1	9999	1	Werkzeug aus Spindel entnommen

Quittungsschritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
xxx	DB4200.DBXx.x	0	6	105

Nach der Zwischenquittung der leeren Spindel wird nach Einsetzten des neuen Werkzeugs in die Spindel der Wechsel mit einer Total-Quittung beendet:

Quittungsschritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
	DB4200.DBX0.0			(99)

Ausgangsposition 2

Handwerkzeug ist in der Spindel und ein Werkzeug aus dem Magazin soll eingewechselt werden

Auftrag von Werkzeugverwaltung an PLC-Anwenderprogramm (Werkzeugwechsel):

DB4300.DBX0.0, DBX1.2 und DBX1.6 ("Werkzeugwechsel vorbereiten")

DB43xx.DBW6	Quell-Magazinnummer	1
DB43xx.DBW8	Quell-Platznummer	6
DB43xx.DBW10	Ziel-Magazinnummer	9999
DB43xx.DBW12	Ziel-Platznummer	1

Quittung "Werkzeugwechsel vorbereiten":

Quittungs-schritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
xxx	DB4200.DBXx.x	0	0	1

Damit steht auf der Nahtstelle:

DB4300.DBX0.0 /1.1 und 1.6 ("Werkzeugwechsel durchführen")

Der Auftrag bleibt sonst unverändert, die Werkzeuge sind noch auf den Ausgangspositionen.

Synchrone Quittung: das Altwerkzeug befindet sich nicht mehr in der Spindel.

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
6 DB9900.DBW40	9998	1	9999	1	Asynchrone Mitteilung, Werkzeug von Spindel entladen

Quittungs-schritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
xxx	DB4200.DBXx.x	0	6	105

Die Spindel ist jetzt leer, das Altwerkzeug ist außerhalb des Magazins.

Nächster Schritt: synchrone Quittung Neuwerkzeug nach Greifer 1

Transferschritt	von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
3 DB9900.DBW16	0	1	9998	2	Neuwerkzeug nach Greifer 1

Quittungs-schritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
xxx	DB4200.DBXx.x	3	0	105

Auftrag ist unverändert.

Nächster Schritt: synchrone Quittung Neuwerkzeug von Greifer 1 nach Spindel:

Transferschritt	von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
4 DB9900.DBW24	9998	2	9998	1	Neuwerkzeug von Greifer 1 nach Spindel

Quittungs-schritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
xxx	DB4200.DBXx.x	4	0	105

Damit sind die Werkzeugbewegungen abgeschlossen.

Ende-Quittung:

Quittungs-schritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
xxx	DB4200.DBXx.x	0	0	1

Der Schritt Neuwerkzeug von Greifer 1 nach Spindel kann entfallen und durch eine Total-Quittung ersetzt werden. Damit wird der Werkzeugverwaltung ebenfalls mitgeteilt, dass alle Werkzeuge auf ihren Zielpositionen sind.

Quittungs-schritt	Quittungsbit	Transferschritt neues WZ	Transferschritt altes WZ	Status
	DB4200.DBX0.0			(99)

10.7 Anwender-Beispiel für Drehmaschine

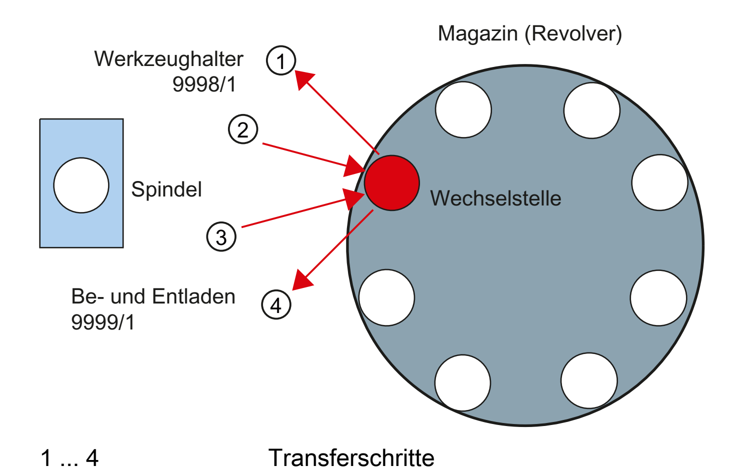
10.7.1 Beispiel: Drehmaschine mit Revolvermagazin (MAG_CONF_MPF)

Beispiel-Datei

Das Programm zur Magazinkonfiguration finden Sie auf der Toolbox.

Das Programm kann in die Steuerung eingelesen werden und ist an die konkrete Anlage anzupassen.

Konfiguration



1 ... 4 Transferschritte

Bild 10-3 Drehmaschine mit Revolvermagazin

Beschreibung des Programms

Zu Beginn werden alle alten Magazindefinitionen und Werkzeuge gelöscht. Im weiteren Programmablauf werden alle Magazine und Zwischenspeicher durch Schreiben der Magazinparameter neu erzeugt und definiert.

In N70 kann die Suchstrategie für Werkzeuge und Magazinplätze festgelegt werden.

Für Revolvermagazine ist es sinnvoll alle Plätze festplatz-codiert zu definieren. Für Magazintyp 3 wird in N320 $TC_MAP3[NUM_MAG] = 81 (Bit 6 = 1) eingestellt.

Die Plätze des Kettenmagazins werden ab N430 bis N500 eingerichtet. Magazin Platztyp = 0 bedeutet, dass der Magazinplatz mit Werkzeugen unterschiedlicher Platztypen beladen werden kann.

Ab N520 werden die Zwischenspeicher eingestellt.

Ab N920 wird die Zuordnung der Zwischenspeicher zur Spindel/Werkzeughalter und dem Magazin vorgenommen.

Nachdem das Programm zur Magazinkonfiguration durchlaufen ist, führen Sie einen Neustart der NC durch (NCK Reset).

Siehe auch

Eine genaue Beschreibung der verwendeten Parameter finden Sie im Funktionshandbuch Werkzeugverwaltung der SINUMERIK 840D sl.

Beispiel MAG_CONF_MPF

1. Anlagenkonfiguration:
   – 1 Revolvermagazin mit 8 Plätzen (in N40 einstellbar)
   – 1 Beladestelle
   – 3 Zwischenspeicherplätze (in N50 einstellbar, ab N540 Zuordnungen)
2. Teileprogramm:

```
;MAG_CONF_MPF
N10 def int NUM_MAG,MAG_TYPE, LOCATIONS,
PLACE, NUM_BUFFER, NUM_LOAD, PLACE_SEARCH
;
N20 NUM_MAG = 1                                ;Nummer des Magazins
N30 MAG_TYPE = 3                               ;Magazinart (1: Kette, 3: Revolver,
                                               5: Flächenmagazin)
N40 LOCATIONS = 8                              ;Anzahl der Magazinplätze
N50 NUM_BUFFER = 1                             ;Anzahl der Zwischenspeicher
                                               (Spindel, Greifer)
N60 NUM_LOAD = 1                               ; Anzahl der Beladestellen
N70 PLACE_SEARCH = 257                         ; Art der Suchstrategie
;= 257 Bit13=0 kein Tausch Altwerkzeug auf Platz von Neuwerkzeug
;Einstellung für Pickup Magazin
;=12289 Bit13=1 Tausch Altwerkzeug auf Platz von Neuwerkzeug
;Einstellung für Kettenmagazin
N80;
N90;
;Parameter prüfen
N100 STOPRE
N110 if ((NUM_MAG==0)or(LOCATIONS==0))
N120 Err1:STOPRE
N130 MSG("Wrong Parameter --> Cancel")
N140 G04 F4
N150 STOPRE
N160 M0
N170 GOTOB Err1
```

```
N180 endif
N190; Magazin Konfiguration
N200;
N210;
N220; Lösche alte Daten wenn, Magazin 1 angelegt wird
N230 if NUM_MAG ==1
N240 $TC_MAP1[0]=0                        ; lösche Magazine
N250 $TC_DP1[0,0]=0                       ; lösche Werkzeuge
N260 STOPRE
N270 endif
; Konfiguration
;
N280 $TC_MAMP2= PLACE_SEARCH              ; Art der Suchstrategie
;
; Magazin
; Magazin einrichten
N290 $TC_MAP1[NUM_MAG]= MAG_TYPE
N300 $TC_MAP2[NUM_MAG]="MAGAZIN"<<NUM_MAG
N310 if MAG_TYPE == 3
N320 $TC_MAP3[NUM_MAG]=81                 ; Magazinzustand, alle Plätze
                                          festplatz-codiert bei
                                          Revolvermagazin
N330 else
N340 $TC_MAP3[NUM_MAG]=17                 ; Magazinzustand
N350 endif
N360 $TC_MAP4[NUM_MAG]=-1
N370 $TC_MAP5[NUM_MAG]=-1
N380 $TC_MAP6[NUM_MAG]=1                  ; Anzahl Zeilen des Magazins
N390 $TC_MAP8[NUM_MAG]=0
N400 $TC_MAP9[NUM_MAG]=0
N410 $TC_MAP7[NUM_MAG]=LOCATIONS          ; Anzahl Magazinplätze
N420 $TC_MAP10[NUM_MAG]=PLACE_SEARCH
;
; Magazinplätze
N430 for PLACE=1 to LOCATIONS
N440 STOPRE
N450 $TC_MPP1[NUM_MAG,PLACE]=1            ; Platzart
N460 $TC_MPP2[NUM_MAG,PLACE]=0            ; Platzytp,0 mit jedem WZ Platztyp
                                          verträglich
N470 $TC_MPP3[NUM_MAG,PLACE]=1            ; Nebenplatzbetrachtung ein (aus
                                          wäre 0)
N480 $TC_MPP4[NUM_MAG,PLACE]=2            ; Platzzustand
N490 $TC_MPP5[NUM_MAG,PLACE]=PLACE        ; Platzartindex
N500 endfor
```

```
N510 STOPRE
;
N520; Definition Zwischenspeichermagazin (immer Nummer 9998)
;
N530 $TC_MAP1[9998]=7                        ; Magazinart 7: Zwischenspeicher
N540 $TC_MAP2[9998]="BUFFER"<<NUM_MAG
N550 $TC_MAP3[9998]=17                       ; Magazinzustand
N560 $TC_MAP6[9998]=1                        ; Anzahl Zeilen
N570 $TC_MAP7[9998]=NUM_BUFFER               ; Anzahl Plätze
;
; Plätze des Zwischenspeichers
; Spindel
N580 $TC_MPP1[9998,1]=2                      ; Platzart (hier Spindel)
N590 $TC_MPP2[9998,1]=0                      ; Platztyp (hier immer 0)
N600 $TC_MPP3[9998,1]=0                      ; Nebenplatzbetrachtung aus
N610 $TC_MPP4[9998,1]=2                      ; Platzzustand
N620 $TC_MPP5[9998,1]=1                      ; Platzartindex
;
N630; Greifer
N640 FOR PLACE=2 to NUM_BUFFER
N650 STOPRE
N660 $TC_MPP1[9998,PLACE]=3                  ;(hier Greifer)
N670 $TC_MPP2[9998,PLACE]=0                  ;(hier immer 0)
N680 $TC_MPP3[9998,PLACE]=0                  ;Nebenplatzbetrachtung aus
N690 $TC_MPP4[9998,PLACE]=2                  ;Platzzustand
N700 $TC_MPP5[9998,PLACE]=PLACE              ;Platzartindex
N710 endfor
N720 STOPRE
;
;
N730; Definition Belademagazin (immer Nummer 9999)
;
N740 $TC_MAP1[9999]=9                        ; Magazinart 9: Belademagazin
N750 $TC_MAP2[9999]="BELADEMAGAZIN"<<NUM_MAG
N760 $TC_MAP3[9999]=17                       ; Magazinzustand
N770 $TC_MAP4[9999]=-1
N780 $TC_MAP5[9999]=-1
N790 $TC_MAP6[9999]=1                        ; Anzahl Zeilen
N800 $TC_MAP7[9999]=NUM_LOAD                 ; Anzahl Plätze
N810 STOPRE;
;
N820; Belademagazinplätze
;
N830 for PLACE=1 to NUM_LOAD
```

```
N840 STOPRE
N850 $TC_MPP1[9999,PLACE]=7                 ; Platzart Beladestelle
N860 $TC_MPP2[9999,PLACE]=0                 ; Platztyp (hier immer 0)
N870 $TC_MPP3[9999,PLACE]=0                 ; Nebenplatzbetrachtung aus
N880 $TC_MPP4[9999,PLACE]=2                 ; Platzzustand
N890 $TC_MPP5[9999,PLACE]=PLACE             ; Platzartindex
N900 endfor
N910 STOPRE
;
;
N920; Offsets (Abstände) ; Abstände zum Magazin
;
; Zwischenspeicher
N930 for PLACE=1 to NUM_BUFFER
N940 $TC_MDP2[1,PLACE]=0
N950 endfor
N960 STOPRE
;
; Beladestellen
N970 for PLACE=1 to NUM_LOAD
N980 stopre
N990 $TC_MDP1[1,PLACE]=0
N1000 endfor

N1010 M30                                   ; Ende
```

Anzeige in der Bediensoftware

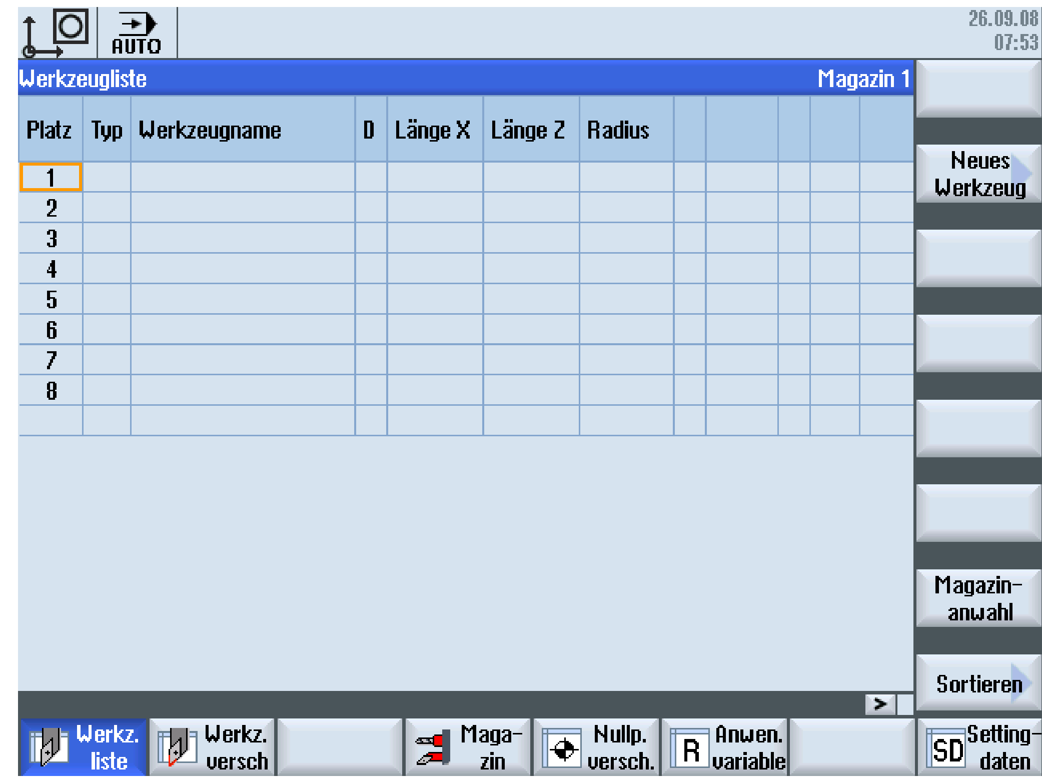


Bild 10-4 Werkzeugliste Drehmaschine

10.7.2 Beispiel: Quittungsschritte (Drehmaschine)

Quittungsschritte

In der Regel sind die mechanischen Abläufe an einer Drehmaschine einfacher als an einer Fräsmaschine. Bei der im vorherigen Kapitel beschriebenen Konfiguration ohne zusätzlichen Zwischenspeicher können die Werkzeugwechsel nach den mechanischen Bewegungen ohne Transferschritte quittiert werden. Das PLC-Anwenderprogramm muss in den Auftragsschnittstellen eingehende Aufträge erkennen und die mechanischen Bewegungen ausführen.

Total-Quittung an Werkzeugverwaltung:

Quittungsschritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
--	DB4000.DBX0.0	--	--	(99)
--	DB4200.DBX0.0	--	--	(99)

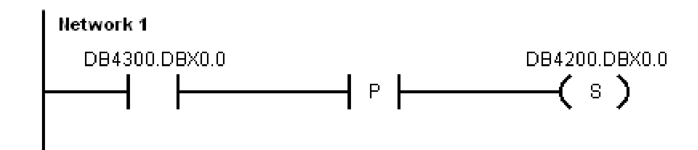


Bild 10-5 Programmierung im PLC-Anwenderprogramm

Für Magazinbewegungen ohne Auftrag von der Werkzeugverwaltung kann eine asynchrone Mitteilung genutzt werden.

Quittung an Werkzeugverwaltung:

Quittungsschritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
xxx	DB4000.DBXx.x	101	0	204

Transferschritt	Quelle		Ziel		Bemerkung
	Magazin	Platz	Magazin	Platz	
101 DB9901.DBW0	1	**n**	9998	1	Der variable Platz im Magazin 1 wurde an die Wechselstelle zur Spindel positioniert.

n ist hier die tatsächliche vom PLC-Anwenderprogramm in die variable Transfertabelle einzutragende Platznummer (n ≠ 0).

10.7.3 Beispiel: Werkzeugwechselzyklus für Drehmaschine

Übergabe-Variablen

Mit MD10717: $MN_T_NO_FCT_CYCLE_NAME (Name des Zyklus, der anstelle der T-Funktion ausgeführt werden soll z. B. "TCHANGE") können Sie einstellen, dass mit Aufruf des T-Befehls ein Zyklus durchlaufen wird. Der Zyklus wird bei jedem T-Aufruf durchlaufen, unabhängig ob ein neues Werkzeug oder das bereits aktive Werkzeug aufgerufen wird. In diesem Zyklus kann die Position des Revolvers auf das programmierte Werkzeug gestellt werden (POSM). Das ist notwendig wenn nach einer Werkzeuganwahl der Revolver von Hand positioniert und das Werkzeug nicht abgewählt wurde. In diesem Fall gibt die NC keinen neuen Auftrag für Werkzeugwechsel durchführen an der Nahtstelle aus.

Programmbeispiel

Voraussetzung ist, dass jede Magazinbewegung der Werkzeugverwaltung mitgeteilt wurde. Das Beispiel ist für Maschinendaten mit der Voreinstellung für Technologie Drehen erstellt. Die Option "Ersatzwerkzeuge für Werkzeugverwaltung" ist nicht aktiv.

Übergabe-Variablen des T-Ersetzungszyklus:

Variable	Beschreibung
$SC_T	T-Nummer des Werkzeugs (numerisch)
$SC_T_Prog	Bool-Variable, die anzeigt, ob in $C_T ein T-Wort zur Verfügung steht.
$C_TS	Bezeichner für Werkzeug (String)
$C_TS_Prog	Bool-Variable, die anzeigt, ob in $C_TS ein Bezeichner zur Verfügung steht.
$C_TE	Adresserweiterung des T-Wortes
$C_D	Programmierte D-Nummer
$C_D_Prog	Bool-Variable, die anzeigt, ob in $C_D eine Korrektur-Nr. zur Verfügung steht.
$C_DL	Programmierte Summen-/Einrichte-Korrektur
$C_DL_Prog	Bool-Variable, die anzeigt, ob in $C_DL eine Korrektur-Nr. zur Verfügung steht.

Im folgenden Beispiel wird mit POSM ein Auftrag zum Positionieren des Magazins an der Nahtstelle ausgegeben. Während der Ausführung der Magazinpositionierung muss die Satzabarbeitung der NC vom PLC-Anwenderprogramm gesteuert werden. In den meisten Fällen wird das Setzen der Einlese- oder Vorschubsperre während dieser Zeit sinnvoll sein. Sobald die im Auftrag angegebene Sollposition erreicht ist (asynchrone Mitteilung der Magazinposition) wird der Auftrag mit Total-Quittung beendet.

```
PROC L6 SAVE SBLOF DISPLOF
  IF $C_T_PROG==1                    ; T ist numerisch
    IF $C_T==0                       ; T=0
      T=0
    ENDIF
    IF $C_T>0
      IF $C_T<=$TC_MAP7[1]           ; Magazinplatz existiert?
      POSM($C_T)                     ; Magazin positionieren
      ENDIF
      T=$C_T                         ; T-Programmierung Platznummer
    ENDIF
  ENDIF

  IF $C_TS_PROG==1                   ; T ist Bezeichner
    _TNO_NEW=GETT($C_TS,1)           ; Abfrage der T-Nummer
    IF _TNO_NEW>0                    ; T-Nummer existiert?
      _TL_NEW=$A_MYMLN[_TNO_NEW]     ; Abfrage Platznummer
    ENDIF
    IF _TL_NEW>0                     ; Werkzeug ist im Magazin?
      POSM(_TL_NEW)                  ; Magazin positionieren
    ENDIF
    T=$C_TS                          ; T-Programmierung ohne Adresserweiterung
```

```
  ENDIF
M17
```

Siehe auch

Die Funktion steht unabhängig von der Werkzeugverwaltung zur Verfügung und ist umfassend beschrieben in: Funktionshandbuch Grundfunktionen, Kapitel "BAG, Kanal, Programmbetrieb, Reset-Verhalten (K1)".

10.7.4 Beispiel: Drehmaschine mit Gegenspindel

Magazinkonfiguration

In der Magazinkonfiguration erhält das Magazin doppelt so viele Plätze wie real vorhanden, z. B. bei einem Revolver mit 12 Plätzen werden 24 Plätze eingerichtet. Platz 1-12 für die Hauptspindel, Platz 13-24 für die Gegenspindel.

Vom PLC-Anwenderprogramm wird das Magazin so positioniert, dass zum Beispiel für Platz 1 und Platz 13 die gleiche Position angefahren wird. Jeder reale Magazinplatz entspricht damit einem virtuellen Magazinplatz für die Hauptspindel und einem virtuellen Magazinplatz für die Gegenspindel.

10.7.5 Beispiel: Test auf leere Zwischenspeicher

Vorgehensweise

Lesen der T-Nummer eines Werkzeuges in Greifer 1 und Greifer 2:

1. Tragen Sie im PLC-Anwenderprogramm die Parameter in DB1200 ein.
2. Setzen Sie in DB1200.DBX0.0 den Start zum Lesen der Platzzustände.

Wurde der Auftrag erfolgreich durchgeführt, erhalten Sie die Ergebnisse ab DB1200.DBB3000:

Variable aus NCK lesen	Adresse	Signal	Werte
Parameter	DB1200.DBW1000	Variablen Index	9
	DB1200.DBW1002	Platznummer	2
	DB1200.DBW1004	Magazinnummer	9998
	DB1201.DBW1000	Variablen Index	9
	DB1201.DBW1002	Platznummer	3
	DB1201.DBW1004	Magazinnummer	9998

Variable aus NCK lesen	Adresse	Signal	Werte
Auftrag	DB1200.DBX0.1	Variable schreiben	0
	DB1200.DBX0.2	PI-Dienst	0
	DB1200.DBB1	Anzahl der Variablen	2
	DB1200.DBX0.0	Start	→ 1
Ergebnis	DB1200.DBB2000.0	Auftrag beendet	1
	DB1200.DBX2000.1	Fehler im Auftrag	0
	DB1200.DBX3000.0	Variable ist gültig	1
	DB1200.DBB3001	Zugriffergebnis	0
	DB1200.DBW3004	Daten von NCK-Variable	n
	DB1201.DBX3000.0	Variable ist gültig	1
	DB1201.DBB3001	Zugriffergebnis	n
	DB1201.DBW3004	Daten von NCK-Variable	0

10.7.6 Beispiel: Werkzeug aus einem Zwischenspeicher ins Magazin transportieren

Vorgehensweise

Ein Werkzeug aus einem Zwischenspeicher (zum Beispiel einem Greifer) soll in das Magazin abgelegt werden. Mit dem PI-Dienst TMMVTL wird die Leerplatzsuche für das Werkzeug aus Greifer 1 (Magazin 9998, Platz 2) ausgeführt und ein Auftrag zum Umsetzen des Werkzeugs erzeugt.

Tragen Sie im PLC-Anwenderprogramm die Parameter in DB1200 ein und setzen Sie in DB1200.DBX4000.0 das Start-Signal des PI-Dienstes.

PI-Dienste im NCK Bereich starten	Adresse	Signal	Werte
Parameter	DB1200.DBW4004	Werkzeugnummer	0
	DB1200.DBW4006	Quell-Platznummer	2
	DB1200.DBW4008	Quell-Magazinnummer	9998
	DB1201.DBW4010	Ziel-Platznummer	-1
	DB1201.DBW4012	Ziel-Magazinnummer	1
Auftrag	DB1200.DBB4001	PI-Index	5
	DB1200.DBX4000.0	Start	→ 1
Ergebnis	DB1200.DBB5000.0	Auftrag beendet	1
	DB1200.DBX5000.1	Fehler im Auftrag	0

Sie erhalten in DB410x.DBX0.0 einen Auftrag zum Umsetzen des Werkzeuges aus dem Zwischenspeicher. Der Zielplatz in Magazin 1 steht in DB4100.DBW12. Das PLC-Anwenderprogramm kann damit den notwendigen Ablauf ausführen.

10.7.7 Beispiel: Auftrag "Werkzeugwechsel vorbereiten" wiederholen

Ablauf: Auftrag wiederholen

Bei einer Fräsmaschine mit Toolboy und Shifter können sich Mitteilung des asynchronen Transfers und Auftrag kreuzen:

- Für das alte Werkzeug wurde mit der Ende-Quittung als Zielposition der Toolboy an die Werkzeugverwaltung quittiert.
- Das Teileprogramm wird mit dem neuen Werkzeug fortgesetzt und benötigt nach sehr kurzer Zeit wieder das vorherige (alte) Werkzeug.
- Die Werkzeugverwaltung generiert den nächsten Vorbereitungsauftrag zum Werkzeugwechsel mit dem Toolboy als Quellposition für das dann neue Werkzeug.
- In der gleichen Zeit wurde vom PLC-Anwenderprogramm das Werkzeug aus dem Toolboy in den Shifter übernommen.
- Das PLC-Anwenderprogramm teilt asynchron die Werkzeugbewegung vom Toolboy in den Shifter mit und beginnt mit dem Weitertransport ins Magazin.
- Im nächsten Takt erkennt das AWP den neuen Auftrag, das Werkzeug vom Toolboy in die Spindel einzuwechseln.

Im Tool-Boy befindet sich aber kein Werkzeug mehr! Das PLC-Anwenderprogramm muss diesen Fall erkennen (Überwachung: enthält Tollboy oder Shifter ein Werkzeug?). Es kann jetzt den Rücktransport des Werkzeugs ins Magazin abbrechen. Von der Werkzeugverwaltung kann dann eine Auftragswiederholung (Status 7) angefordert werden:

- Die Werkzeugverwaltung hat inzwischen die Mitteilung erhalten, dass sich das Werkzeug im Shifter befindet.
- Sie generiert einen neuen Vorbereitungsauftrag mit der Quellposition Shifter für das neue Werkzeug.

Hinweis

Die Quittung zur Wiederholung eines Auftrags "Werkzeugwechsel vorbereiten" darf nur **vor** der Ende-Quittung des Auftrags "Werkzeugwechsel vorbereiten" erfolgen!

Quittung an Werkzeugverwaltung:

Quittungsschritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
xxx	DB4000.DBXx.x	0	0	7

10.8 Anwender-Beispiel für Fräsmaschine

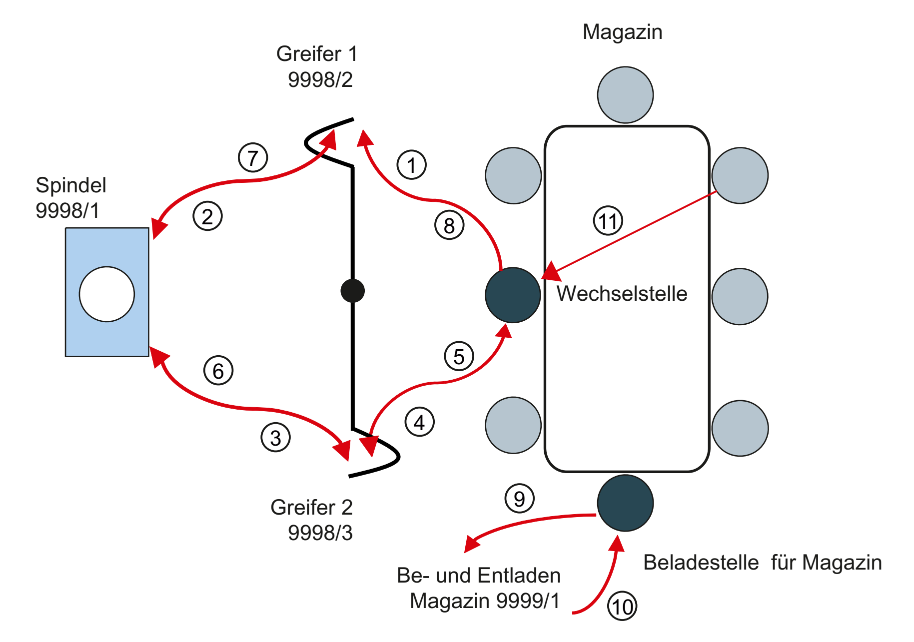
10.8.1 Beispiel: Fräsmaschine mit Kettenmagazin und Doppelgreifer (MAG_CONF_MPF)

Beispiel-Datei

Das Programm zur Magazinkonfiguration finden Sie auf der Toolbox.

Das Programm kann in die Steuerung eingelesen werden und ist an die konkrete Anlage anzupassen.

Konfiguration



1 ... 11 Transferschritte

Bild 10-6 Fräsmaschine mit Kettenmagazin

Bezeichnung der Zwischenspeicher und Beladestellen:

Magazin	Platz	Bedeutung
1	xx	reales Magazin (Kette, Teller, Fläche), Position xx
9998	1	Spindel
9998	2	Greifer 1
9998	3	Greifer 2
9999	1	Beladestelle für Magazin

Beschreibung des Programms

Zu Beginn werden alle alten Magazindefinitionen und Werkzeuge gelöscht. Im weiteren Programmablauf werden alle Magazine und Zwischenspeicher durch Schreiben der Magazinparameter neu erzeugt und definiert.

In N70 kann die Suchstrategie für Werkzeuge und für Magazinplätze ausgewählt werden. Hier wird festgelegt ob ein Werkzeug bei Wechsel aus der Spindel direkt auf den Platz des Neuwerkzeugs abgelegt werden kann. Damit ist der Werkzeugwechsel in einem mechanischen Ablauf möglich, es können so kürzere Wechselzeiten erreicht werden. An Pickup-Magazinen ist dieser Ablauf nicht möglich.

Die Plätze des Kettenmagazins werden ab N430 bis N500 eingerichtet. Magazin Platztyp = 0 bedeutet das der Magazinplatz mit Werkzeugen unterschiedlicher Platztypen beladen werden kann.

Ab N520 werden die Zwischenspeicher eingestellt. Sind zusätzliche Zwischenspeicher vorhanden (Toolboy, Shifter …) ist die Anzahl in N50 zu ändern.

Analog dazu ist bei zusätzlich vorhandenen Beladestellen in N60 zu verfahren.

Ab N920 wird die Zuordnung der Zwischenspeicher zur Spindel/Werkzeughalter und dem Magazin vorgenommen.

Nachdem das Programm zur Magazinkonfiguration durchlaufen ist führen Sie einen Neustart der NC durch (NCK Reset).

Siehe auch

Weitere Informationen:

- Die Konfiguration eines Tellermagazins mit Festplatz-Codierung erfolgt wie die Konfiguration einer Revolvermagazins: Beispiel: Drehmaschine mit Revolvermagazin (MAG_CONF_MPF) (Seite 360)
- Eine genaue Beschreibung der verwendeten Parameter finden Sie im Funktionshandbuch Werkzeugverwaltung der SINUMERIK 840D sl.

Beispiel MAG_CONF_MPF

1. Anlagenkonfiguration:
   - 1 Kettenmagazin mit 8 Plätzen (in N40 einstellbar)
   - 1 Beladestelle
   - 3 Zwischenspeicherplätze (in N50 einstellbar, ab N540 Zuordnungen)
2. Teileprogramm:

```
;MAG_CONF_MPF
N10 def int NUM_MAG,MAG_TYPE, LOCATIONS,
PLACE, NUM_BUFFER, NUM_LOAD, PLACE_SEARCH
;
N20 NUM_MAG = 1                                 ;Nummer des Magazins
```

```
N30 MAG_TYPE = 1                        ;Magazinart (1: Kette, 3: Revolver,
                                        5: Flächenmagazin)
N40 LOCATIONS = 8                       ;Anzahl der Magazinplätze
N50 NUM_BUFFER = 3                      ;Anzahl der Zwischenspeicher
                                        (Spindel, Greifer)
N60 NUM_LOAD = 1                        ; Anzahl der Beladestellen
N70 PLACE_SEARCH = 12289                ; Art der Suchstrategie
;= 257 Bit13=0 kein Tausch Altwerkzeug auf Platz von Neuwerkzeug
;Einstellung für Pickup Magazin
;=12289 Bit13=1 Tausch Altwerkzeug auf Platz von Neuwerkzeug
;Einstellung für Kettenmagazin
N80;
N90;
;Parameter prüfen
N100 STOPRE
N110 if ((NUM_MAG==0)or(LOCATIONS==0))
N120 Err1:STOPRE
N130 MSG("Wrong Parameter --> Cancel")
N140 G04 F4
N150 STOPRE
N160 M0
N170 GOTOB Err1
N180 endif
N190; Magazin Konfiguration
N200;
N210;
N220; Lösche alte Daten wenn, Magazin 1 angelegt wird
N230 if NUM_MAG ==1
N240 $TC_MAP1[0]=0                      ; lösche Magazine
N250 $TC_DP1[0,0]=0                     ; lösche Werkzeuge
N260 STOPRE
N270 endif
; Konfiguration
;
N280 $TC_MAMP2= PLACE_SEARCH            ; Art der Suchstrategie
;
; Magazin
; Magazin einrichten
N290 $TC_MAP1[NUM_MAG]= MAG_TYPE
N300 $TC_MAP2[NUM_MAG]="MAGAZIN"<<NUM_MAG
N310 if MAG_TYPE == 3
N320 $TC_MAP3[NUM_MAG]=81               ; Magazinzustand, alle Plätze
                                        festplatz-codiert bei
                                        Revolvermagazin
```

```
N330 else
N340 $TC_MAP3[NUM_MAG]=17                      ; Magazinzustand
N350 endif
N360 $TC_MAP4[NUM_MAG]=-1
N370 $TC_MAP5[NUM_MAG]=-1
N380 $TC_MAP6[NUM_MAG]=1                       ; Anzahl Zeilen des Magazins
N390 $TC_MAP8[NUM_MAG]=0
N400 $TC_MAP9[NUM_MAG]=0
N410 $TC_MAP7[NUM_MAG]=LOCATIONS               ; Anzahl Magazinplätze
N420 $TC_MAP10[NUM_MAG]=PLACE_SEARCH
;
; Magazinplätze
N430 for PLACE=1 to LOCATIONS
N440 STOPRE
N450 $TC_MPP1[NUM_MAG,PLACE]=1                 ; Platzart
N460 $TC_MPP2[NUM_MAG,PLACE]=0                 ; Platzytp,0 mit jedem WZ Platztyp
                                               verträglich
N470 $TC_MPP3[NUM_MAG,PLACE]=1                 ; Nebenplatzbetrachtung ein (aus
                                               wäre 0)
N480 $TC_MPP4[NUM_MAG,PLACE]=2                 ; Platzzustand
N490 $TC_MPP5[NUM_MAG,PLACE]=PLACE             ; Platzartindex
N500 endfor
N510 STOPRE
;
N520; Definition Zwischenspeichermagazin (immer Nummer 9998)
;
N530 $TC_MAP1[9998]=7                          ; Magazinart: 7: Zwischenspeicher
N540 $TC_MAP2[9998]="BUFFER"<<NUM_MAG
N550 $TC_MAP3[9998]=17                         ; Magazinzustand
N560 $TC_MAP6[9998]=1                          ; Anzahl Zeilen
N570 $TC_MAP7[9998]=NUM_BUFFER                 ; Anzahl Plätze
;
; Plätze des Zwischenspeichers
; Spindel
N580 $TC_MPP1[9998,1]=2                        ; Platzart (hier Spindel)
N590 $TC_MPP2[9998,1]=0                        ; Platztyp (hier immer 0)
N600 $TC_MPP3[9998,1]=0                        ; Nebenplatzbetrachtung aus
N610 $TC_MPP4[9998,1]=2                        ; Platzzustand
N620 $TC_MPP5[9998,1]=1                        ; Platzartindex
;
N630; Greifer
N640 FOR PLACE=2 to NUM_BUFFER
N650 STOPRE
N660 $TC_MPP1[9998,PLACE]=3                    ;(hier Greifer)
```

```
N670 $TC_MPP2[9998,PLACE]=0                    ;(hier immer 0)
N680 $TC_MPP3[9998,PLACE]=0                    ;Nebenplatzbetrachtung aus
N690 $TC_MPP4[9998,PLACE]=2                    ;Platzzustand
N700 $TC_MPP5[9998,PLACE]=PLACE                ;Platzartindex
N710 endfor
N720 STOPRE
;
;
N730; Definition Belademagazin (immer Nummer 9999)
;
N740 $TC_MAP1[9999]=9                          ; Magazinart: 9: Belademagazin
N750 $TC_MAP2[9999]="BELADEMAGAZIN"<<NUM_MAG
N760 $TC_MAP3[9999]=17                         ; Magazinzustand
N770 $TC_MAP4[9999]=-1
N780 $TC_MAP5[9999]=-1
N790 $TC_MAP6[9999]=1                          ; Anzahl Zeilen
N800 $TC_MAP7[9999]=NUM_LOAD                   ; Anzahl Plätze
N810 STOPRE;
;
N820; Belademagazinplätze
;
N830 for PLACE=1 to NUM_LOAD
N840 STOPRE
N850 $TC_MPP1[9999,PLACE]=7                    ; Platzart Beladestelle
N860 $TC_MPP2[9999,PLACE]=0                    ; Platztyp (hier immer 0)
N870 $TC_MPP3[9999,PLACE]=0                    ; Nebenplatzbetrachtung aus
N880 $TC_MPP4[9999,PLACE]=2                    ; Platzzustand
N890 $TC_MPP5[9999,PLACE]=PLACE                ; Platzartindex
N900 endfor
N910 STOPRE
;
;
N920; Offsets (Abstände) ; Abstände zum Magazin
;
; Zwischenspeicher
N930 for PLACE=1 to NUM_BUFFER
N940 $TC_MDP2[1,PLACE]=0
N950 endfor
N960 STOPRE
;
; Beladestellen
N970 for PLACE=1 to NUM_LOAD
N980 stopre
N990 $TC_MDP1[1,PLACE]=0
```

```
N1000 endfor
N1010 M30                                              ; Ende
```

Anzeige in der Bediensoftware

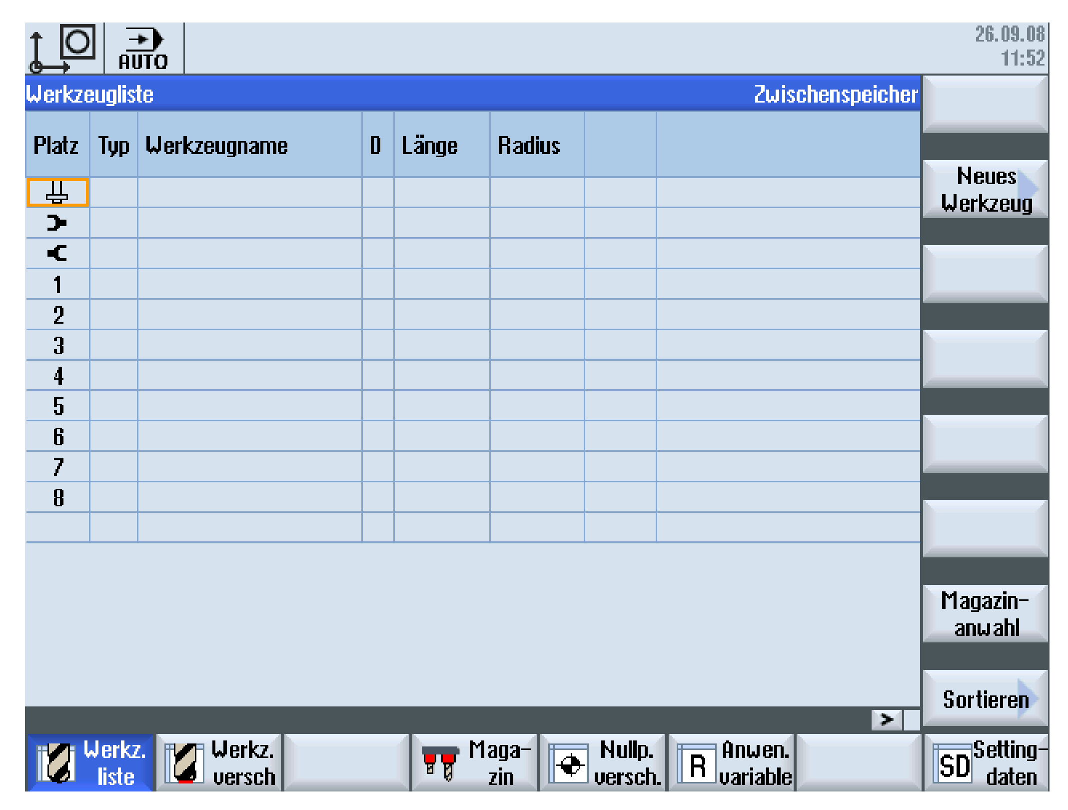


Bild 10-7 Werkzeugliste Fräsmaschine

10.8.2 Ablaufdiagramm: Werkzeugwechsel

Programmablauf Werkzeugwechsel (PLC)

Der hier beschriebene Ablauf beschreibt den Wechsel zwischen Magazin und Spindel. Der Wechsel von Handwerkzeugen sowie Be- und Entladen sind nicht berücksichtigt. Diese Abläufe sind beschrieben in:

- Beispiel: Beladen / Entladen (Seite 354)
- Beispiel: Handwerkzeuge wechseln (Seite 356)

Die Voreinstellung der Maschinendaten ist so gewählt, dass bei Programmierung mit dem T-Befehl an der Nahtstelle der Auftrag für "Werkzeugwechsel vorbereiten" ausgelöst wird:

```
N10 T = "Werkzeugname" M6
```

Der Satzvorlauf wird nicht unterbrochen. Gleichzeitig wird mit M6 das Werkzeugwechsel-Unterprogramm gestartet (L6). Sobald der Auftrag für "Werkzeugwechsel vorbereiten" quittiert wurde und im Werkzeugwechsel Unterprogramm M-Code für Werkzeugwechsel ausgegeben erreicht ist, wird an der Nahtstelle der Auftrag "Werkzeugwechsel durchführen" ausgegeben (Satz-Splitting).

Einem Werkzeugwechselbefehl (M206) muss immer ein Befehl zum Werkzeugwechsel vorbereiten vorangehen. Ein Wechselbefehl ohne vorherigen Auftrag "Werkzeugwechsel vorbereiten" löst keinen Auftrag der Werkzeugverwaltung aus.

Im Programmablauf finden Sie den Ausdruck 1:1 Wechsel. Das bedeutet, dass der Werkzeugwechsel in einem Zyklus durchgeführt wird. Das Werkzeug aus der Spindel (Altwerkzeug) wird auf dem Magazinplatz des Neuwerkzeugs abgelegt. Es ist keine zusätzliche Magazinpositionierung notwendig. In diesem Fall ist im Auftrag der Werkzeugverwaltung der Zielplatz des Altwerkzeugs gleich dem Quellplatz des Neuwerkzeugs (DB43xx.DBW6 und DBW8 sind gleich DBW18 und DBW20).

Ein 1:1 Wechsel ist nicht möglich bei:

- unterschiedlichen Platztypen der Werkzeuge
- unterschiedlichen Werkzeuggrößen
- Festplatzwerkzeugen

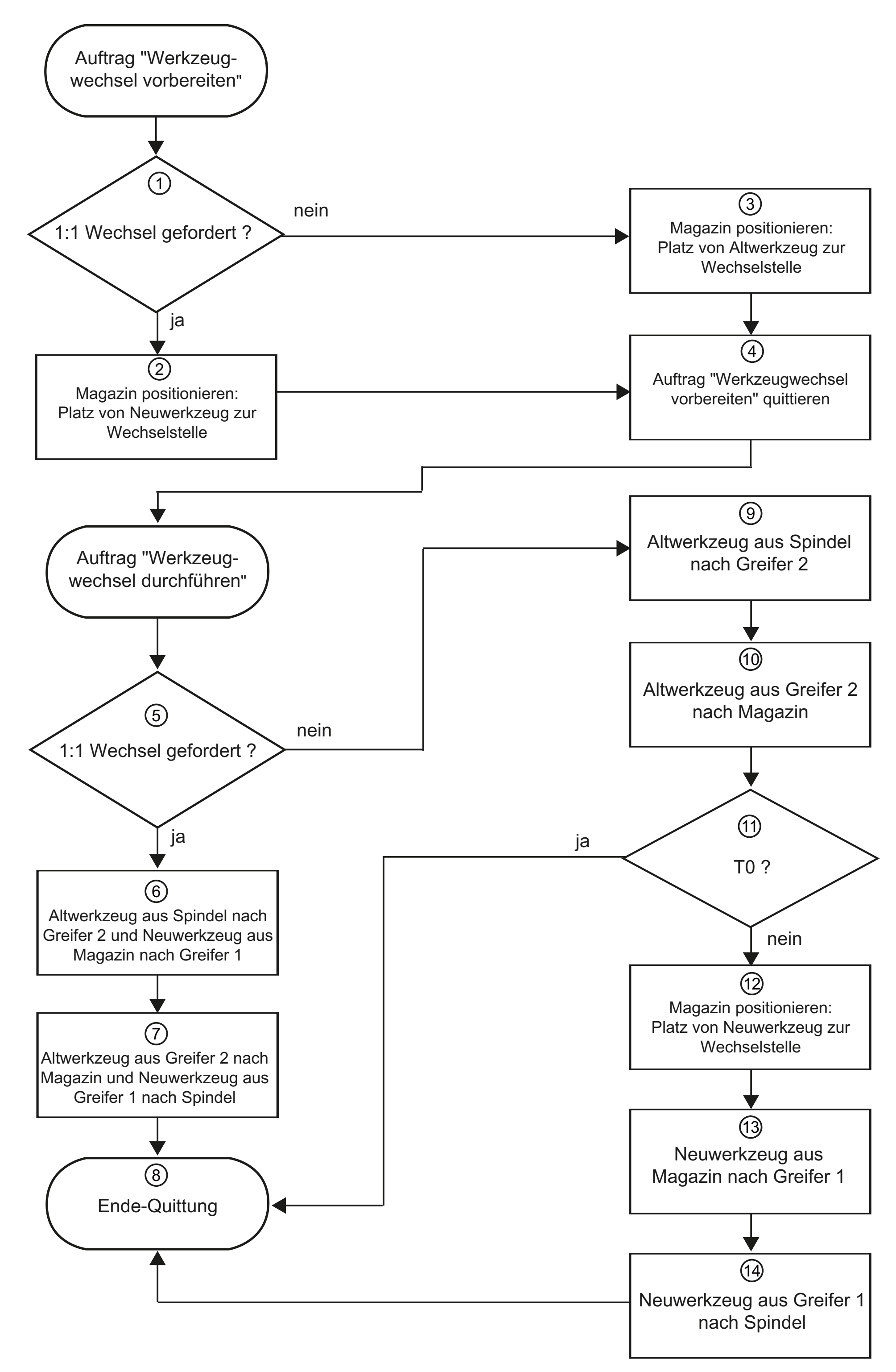
Mit Programmierung von T0 im NC-Programm wird ein Werkzeugwechsel ohne Neuwerkzeug angestoßen. Es wird nur das Spindelwerkzeug ins Magazin transportiert (Spindel leer fahren).

① ... ⑭ bezeichnen die Schritte im folgenden Ablaufdiagramm.

Sie können anhand der Schritte die unterschiedlichen Quittungsmöglichkeiten sehen. Es kommen sowohl auftragsbezogene Quittungen als auch asynchrone Mitteilungen zum Einsatz.

Die programmierten Transferschritte entnehmen Sie bitte der Tabelle in Kapitel Beispiel: Quittungsschritte (Fräsmaschine) (Seite 386)

Ablaufdiagramm "Werkzeugwechsel"



1: 1 Wechsel: Das Altwerkzeug wird auf den Platz des Neuwerkzeugs abgelegt.

Bild 10-8 Ablaufdiagramm

Beschreibung des Ablaufs

- NC Programm:

  T-Befehl oder T-Befehl mit gleichzeitigem WZW-Aufruf (M6)
- Nahtstellensignale:

  Auftrag von WZV: DB43xx.DBX0.0 (Auftragsbit) und

  DB43xx.DBB1 (Kommandobits): Wechsel vorbereiten (DB43xx.DBX1.2)
- Magazin- und Platznummer der zu bewegenden Werkzeuge:

  DB43xx.DBW6 bis DBW20: Quellplatz neues Werkzeug, Zielplatz altes Werkzeug

Schritt 1: 1:1 Wechsel gefordert

Abfrage: Quellplatz neues Werkzeug == Zielplatz altes Werkzeug, Normalfall:

Es befindet sich ein Werkzeug im Werkzeughalter, und es wird ein neues Werkzeug angefordert. Beide Werkzeuge haben den gleichen Platztyp und gleiche Werkzeuggröße in der Magazinliste, die Werkzeuge sind nicht festplatzcodiert.

Das Altwerkzeug wird in direktem Wechsel auf dem Magazinplatz des Neuwerkzeugs abgelegt (1:1 Wechsel). Ist kein Altwerkzeug im Werkzeughalter (DB4300.DBX1.4), erfolgt der gleiche Ablauf. Das Magazin wird in diesem Fall das Neuwerkzeug zur Wechselstelle positionieren.

→ weiter mit Schritt 2

Abfrage: Quellplatz neues Werkzeug >< Zielplatz altes Werkzeug, Sonderfälle:

Über DB43xx.DBX1.3 wird von der WZV mitgeteilt, ob das aktive Werkzeug aus dem Werkzeughalter entfernt werden soll (T0). In diesem Fall wird das Magazin den Ablageplatz des Altwerkzeugs (das eben noch in der Spindel ist) zur Wechselstelle positionieren.

Der Magazinplatz für das Altwerkzeug wird auch dann zur Wechselstelle positioniert, wenn das Altwerkzeug nicht auf dem Platz des Neuwerkzeugs abgelegt werden kann (1:1 Wechsel nicht möglich). Ursache können unterschiedliche Platztypen oder Werkzeuggrößen sein, oder es sind festplatzcodierte Werkzeuge beteiligt. In diesem Fall wird der Werkzeugwechsel in zwei Schritten durchgeführt. Zuerst wird das Altwerkzeug in das Magazin abgelegt und danach das Neuwerkzeug zur Spindel transportiert.

→ weiter mit Schritt 3

Schritt 2: Magazin positionieren, Platz von Neuwerkzeug zur Wechselstelle.

← vorheriger Schritt: Schritt 1

Die Magazinbewegung kann z.B. über eine PLC gesteuerte NC Rundachse erfolgen. Die Bewegung sollte an die Werkzeugverwaltung gemeldet werden. Die Position des Magazins wird damit an der Bedienoberfläche in den Ansichten der Werkzeug- und Magazinliste aktualisiert.

Von DB4300.DBW8 (Platz Nummer für neues Werkzeug - Quelle) wird die Zielposition gelesen und nach DB9901.DBW2 (Transferschritt 101) geschrieben. Bei Magazinkoinzidenz an der Zielposition wird der Schritt asynchron quittiert:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
4	DB4200.DBX0.4	101	0	204

Quittung an WZV:

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
101 DB9901.DBW0	1	n	9998	1	Der variable Platz im Magazin 1 wurde an die Wechselstelle zur Spindel positioniert.

n: ist hier die tatsächliche, vom PLC-Anwenderprogramm in die variable Transfertabelle einzutragende Platznummer (n ≠ 0).

→ weiter mit Schritt 4

Schritt 3: Magazin positionieren, Platz von Altwerkzeug zur Wechselstelle.

← vorheriger Schritt: Schritt 1

Analog zu Schritt 2, jedoch wird der Zielplatz des Magazins hier von DB4300.DBW20 (Platznummer für altes Werkzeug - Ziel) bezogen.

Quittung an WZV:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
4	DB4200.DBX0.4	101	0	204

Transferschritt	von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
101 DB9901.DBW0	1	n	9998	1	Der variable Platz im Magazin 1 wurde an die Wechselstelle zur Spindel positioniert.

n: ist hier die tatsächliche, vom PLC-Anwenderprogramm in die variable Transfertabelle einzutragende Platznummer (n ≠ 0).

→ weiter mit Schritt 4

Schritt 4: Auftrag Werkzeugwechsel vorbereiten quittieren

← vorheriger Schritt: Schritt 2 oder Schritt 3

An vielen Systemen ist damit die Vorbereitung des Werkzeugwechsels abgeschlossen.

Quittung an WZV:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
1	DB4200.DBX0.1	0	0	1

→ **weiter mit Schritt 5**

- NC Programm:

  M206 löst den Auftrag für WZW durchführen aus

- Nahtstellensignale:

  Auftrag von WZV: DB43xx.DBX0.0 (Auftragsbit)

  DB43xx.DBB1 (Kommandobits): Wechsel durchführen (DB43xx.DBX1.1)

- Magazin- und Platznummer der zu bewegenden Werkzeuge:

  DB43xx.DBW6 bis DBW20: Quellplatz neues Werkzeug, Zielplatz altes Werkzeug

Schritt 5: 1:1 Wechsel gefordert

← **vorheriger Schritt: kein vorheriger Schritt, Einsprung in die Schrittkette bei Auftrag "Werkzeugverwaltung durchführen"**

Analog zu Schritt 1 wird geprüft, ob ein direkter Wechsel oder ein Wechsel in zwei Schritten durchgeführt wird:

- 1:1 Wechsel möglich: → **weiter mit Schritt 6**
- 1:1 Wechsel nicht möglich: → **weiter mit Schritt 9**

Schritt 6: Altwerkzeug aus Spindel nach Greifer 2 und Neuwerkzeug aus Magazin nach Greifer 1

← **vorheriger Schritt: Schritt 5**

Das PLC-Programm steuert die Maschinenfunktionen, mit denen die Greiferbewegungen, Werkzeugklemmung usw. stattfinden. Sind die mechanischen Bewegungen abgeschlossen und im PLC-Anwenderprogramm quittiert, erfolgt die Quittung der Werkzeugbewegungen an die Werkzeugverwaltung.

Quittung an WZV:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
5	DB4200.DBX0.5	1	2	105

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
1 DB9900.DBW0	0	1	9998	2	Neu Werkzeug von Magazin nach Greifer 1 Schritt 6 oder 13
2 DB9900.DBW8	9998	1	9998	3	Alt Werkzeug von Spindel nach Greifer 2 Schritt 6 oder 9

→ weiter mit Schritt 7.

Schritt 7: Altwerkzeug aus Greifer 2 nach Magazin und Neuwerkzeug aus Greifer 1 nach Spindel

← vorheriger Schritt: Schritt 6

Das PLC-Programm steuert die Maschinenfunktionen, für die Greiferbewegungen, Werkzeugklemmung usw. Sind die mechanischen Bewegungen abgeschlossen und im PLC-Anwenderprogramm quittiert, erfolgt die Quittung der Werkzeugbewegungen an die Werkzeugverwaltung.

Quittung an Werkzeugverwaltung:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
7	DB4200.DBX0.7	3	4	105

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
3 DB9900.DBW16	9998	2	9998	1	Neu Werkzeug von Greifer 1 nach Spindel Schritt 7 oder 14
4 DB9900.DBW24	9998	3	0	2	Alt Werkzeug von Greifer 2 nach Magazin Schritt 7 oder 10

→ weiter mit Schritt 8

Schritt 8: Ende-Quittung

← vorheriger Schritt: Schritt 7 oder Schritt 14

Mit Werkzeugwechsler in Grundstellung oder einem Zustand, in dem die Maschine die Bearbeitung fortsetzen kann, erfolgt die Ende-Quittung. Es sind hier eventuell noch mechanische Bewegungen durchzuführen, bevor der Werkzeugwechsel beendet werden kann.

Quittung an Werkzeugverwaltung:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
1	DB4200.DBX0.1	0	0	1

→ weiter mit Schritt 9

Schritt 9: Altwerkzeug aus Spindel nach Greifer 2

← vorheriger Schritt: Schritt 5

Das PLC-Programm steuert die Maschinenfunktionen, mit denen die Greiferbewegungen, Werkzeugklemmung usw. stattfinden. Sind die mechanischen Bewegungen abgeschlossen und im PLC-Anwenderprogramm quittiert, erfolgt die Quittung der Werkzeugbewegungen an die Werkzeugverwaltung.

Quittung an Werkzeugverwaltung:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
8	DB4200.DBX1.0	0	2	105

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
2 DB9900.DBW8	9998	1	9998	3	Alt Werkzeug von Spindel nach Greifer 2 Schritt 6 oder 9

→ weiter mit Schritt 10

Schritt 10: Altwerkzeug aus Greifer 2 nach Magazin

← vorheriger Schritt: Schritt 9

Das PLC-Programm steuert die Maschinenfunktionen, mit denen die Greiferbewegungen, Werkzeugklemmung usw. stattfinden. Sind die mechanischen Bewegungen abgeschlossen und im PLC-Anwenderprogramm quittiert, erfolgt die Quittung der Werkzeugbewegungen an die Werkzeugverwaltung.

Quittung an Werkzeugverwaltung:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
9	DB4200.DBX1.1	0	4	105

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
4 DB9900.DBW24	9998	3	0	2	Alt Werkzeug von Greifer 2 nach Magazin Schritt 7 oder 10

→ weiter mit Schritt 11

Schritt 11: T0 ?

← vorheriger Schritt: Schritt 10

Abfrage: ist im Werkzeugwechselauftrag T0 gesetzt?

DB43xx.DBX1.3

Soll nur der Werkzeughalter leer gefahren werden, kann der Werkzeugwechsel beendet werden:

→ weiter mit Schritt 8

Soll ein neues Werkzeug zum Werkzeugträger gebracht werden?

→ weiter mit Schritt 12

Schritt 12: Magazin positionieren Platz von Neuwerkzeug zur Wechselstelle

← vorheriger Schritt: Schritt 11

Ablauf wie Schritt 2

Quittung an Werkzeugverwaltung:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
4	DB4200.DBX0.4	101	0	204

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
101 DB9901.DBW0	1	n	9998	1	Der variable Platz im Magazin 1 wurde an die Wechselstelle zur Spindel positioniert.

n: ist hier die tatsächliche, vom PLC-Anwenderprogramm in die variable Transfertabelle einzutragende Platznummer (n ≠ 0).

→ weiter mit Schritt 13

Schritt 13: Neuwerkzeug aus Magazin nach Greifer 1

← vorheriger Schritt: Schritt 12

Das PLC-Programm steuert die Maschinenfunktionen, mit denen die Greiferbewegungen, Werkzeugklemmung usw. stattfinden.

Quittung an Werkzeugverwaltung:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
10	DB4200.DBX1.2	1	0	105

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
1 DB9900.DBW0	0	1	9998	2	Neu Werkzeug von Magazin nach Greifer 1 Schritt 6 oder 13

→ weiter mit Schritt 14

Schritt 14: Neuwerkzeug aus Greifer 1 nach Spindel

← vorheriger Schritt: Schritt 13

Das PLC-Anwenderprogramm steuert die Maschinenfunktionen, mit denen die Greiferbewegungen, Werkzeugklemmung usw. stattfinden. Sind die mechanischen Bewegungen abgeschlossen und im PLC-Anwenderprogramm quittiert, erfolgt die Quittung der Werkzeugbewegungen an die Werkzeugverwaltung.

Der Werkzeugwechsel kann beendet werden.

Quittung an Werkzeugverwaltung:

Quittungs-schritt	Quittungsbit	Transferschritt neues Werkzeug	Transferschritt altes Werkzeug	Status
11	DB4200.DBX1.3	3	0	105

Transferschritt	Von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
3 DB9900.DBW16	9998	2	9998	1	Neu Werkzeug von Greifer 1 nach Spindel Schritt 7 oder 14

→ weiter mit Schritt 8

10.8.3 Beispiel: Quittungsschritte (Fräsmaschine)

Konstante Transferschritt-Tabelle

Transferschritt	von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
1 DB9900.DBW0	0	1	9998	2	Neues Werkzeug von Magazin nach Greifer 1 Schritt ⑥ oder ⑬
2 DB9900.DBW8	9998		9998	3	Schritt ⑥ oder ⑨
3 DB9900.DBW16	9998		9998	1	Schritt ⑦ oder ⑩
4 DB9900.DBW24	9998	3	0	2	Schritt ⑦ oder ⑭
5 DB9900.DBW32	0	2	9998	1	Ablageplatz des alten Werkzeugs
6 DB9900.DBW40	0	1	9998	1	Magazinplatz mit dem neuen Werkzeug zur Wechselstelle Schritt ② oder ⑫
7 DB9900.DBW48	--	--	--	--	

Variable Transferschritt-Tabelle

Transferschritt	von		nach		Bemerkung
	Magazin	Platz	Magazin	Platz	
101 DB9901.DBW0	1	n	9998	1	Der variable Platz im Magazin 1 wurde an die Wechselstelle zur Spindel positioniert.
102 DB9901.DBW8	--	--	--	--	

n: ist hier die tatsächliche, vom PLC-Anwenderprogramm in die variable Transfertabelle einzutragende Platznummer (n ≠ 0).

Quittungsschritt-Tabelle

Quittungsschritt	Transferschritt		Quittungs-status	Bemerkung
	Altes Werkzeug	Neues Werkzeug		
1 DB9902.DBW0	0	0	1	Ende-Quittung, Schritt ④ und ⑧
2 DB9902.DBW4	0	0	3	Auftrag abbrechen
3 DB9902.DBW8	0	0	105	Zwischenquittung bei Folgeauftrag, Schritt ④
4 DB9902.DBW12	101	0	204	Variabler Magazinplatz zur Wechselstelle
5 DB9902.DBW16	1	2	105	Zwischenquittung Schritt ⑥
6 DB9902.DBW20	0	5	105	Zwischenquittung Schritt ③
7 DB9902.DBW24	3	4	105	Zwischenquittung Schritt ⑦
8 DB9902.DBW28	0	2	105	Zwischenquittung Schritt ⑨
9 DB9902.DBW32	0	4	105	Zwischenquittung Schritt ⑩
10 DB9902.DBW36	1	0	105	Zwischenquittung Schritt ⑬
11 DB9902.DBW40	3	0	105	Zwischenquittung Schritt ⑭
12 DB9902.DBW44	--	--	--	

Hinweis: Die Schrittnummern ① ... ⑭ beziehen sich auf das Ablaufdiagramm im Kapitel Ablaufdiagramm: Werkzeugwechsel (Seite 376)

10.8.4 Beispiel: Werkzeugwechselzyklus für Fräsmaschine

Programmbeispiel

```
PROC L6 SAVE DISPLOF
;----------------------------------------------------------------
; Beispiel Werkzeugwechselzyklus für Maschinenhersteller
;----------------------------------------------------------------
DEF INT _WZ_IN_SP,_WZ_VOR
DEF REAL _SPP= ...                        ; Spindelposition
;
  IF(NOT $P_SEARCH)                       ; wenn kein Satzsuchlauf
    _WZ_IN_SP=$TC_MPP6[9998,1]            ; Werkzeug in der Spindel
    GETSELT(_WZ_VOR)                      ; vorher angewähltes Werkzeug
;
    IF(_WZ_IN_SP<>_WZ_VOR)                ; wenn anderes Werkzeug
      SPOS=_SPP                           ; Spindel positionieren
      G0                                  ; Werkzeugwechselposition anfahren
      G75 Z=0
      WAITS(1)
    ENDIF
  ELSE
  ENDIF
;
; Werkzeug einwechseln: Werkzeugverwaltung und PLC
M206
M17
;----------------------------------------------------------------
; ENDE
;----------------------------------------------------------------
```

11 Easy Archive

11.1 Daten sichern und archivieren

Übersicht

Entsprechend der Aufteilung der Daten in Datenklassen kann getrennt für jeden Datenbereich und für jede Datenklasse ein Archiv erzeugt werden.

Eine Ausnahme bildet die Datenklasse "System": Diese Daten werden fest eingestellt und bei der Erst-Installation oder bei einer Default-Initialisierung wirksam. Eine Datensicherung der Systemdaten ist aus diesem Grund nicht erforderlich, da in dieser Datenklasse keine Daten enthalten sind, die bei der Inbetriebnahme oder zur Laufzeit der Maschine gebildet werden.

Ein von Siemens geliefertes Systemarchiv kann z. B. eine neue NCK-Version oder integrierte HMI-Version sein oder auch ein Zyklen-Hotfix beinhalten.

ACHTUNG
Schutz der Systemdaten Alle Systemdaten und Dateninhalte der Datenklasse "System" in den Bereichen HMI, NCK, PLC und Antriebe dürfen nicht beeinflusst werden. Systemdaten können durch Bedienung oder durch Beschreiben aus einem Teile- oder Unterprogramm oder Zyklus und durch Einlesen eines Archivs nicht verändert werden. Wenn Sie Anwenderdaten mit einem USB-FlashDrive einspielen, darf die Datenmenge nicht größer als 4 MB sein!

Kennzeichnung von Archiven

Jedes Archiv enthält folgende Kennungen:

- Datenklasse: M, I, U
- Steuerungstyp: 828D TE oder 828D ME
- Zeitstempel: Datum und Uhrzeit, wann das Archiv erstellt wurde.
- Versionsbezeichnung: Software-Version, mit der dieses Archiv erstellt wurde.
- Seriennummer der System CompactFlash Card.

Diese Kennzeichnungen erlauben es, beim Einlesen Filter zu setzen, welches Archiv auf welcher Steuerungsvariante in Abhängigkeit der Datenklasse und der Softwareversion eingelesen werden darf.

Kompatibilität der Daten

PLC-Archive von einer SINUMERIK 802D sl können über das Programming Tool auf eine SINUMERIK 828D-Steuerungsvariante übernommen werden.

11.2 Daten sichern in Archiven

Wann sichern Sie Inbetriebnahme-Daten?

Um eine Datensicherung durchzuführen, sind folgende Zeitpunkte empfehlenswert:

- nach der Inbetriebnahme
- nach der Änderung von maschinenspezifischen Einstellungen
- nach dem Austausch einer Hardware-Komponente
- vor einer Software-Hochrüstung

Daten sichern und wiederherstellen

Um Daten zu sichern und wiederherzustellen, wählen Sie im Bedienbereich "Inbetriebnahme":

- Softkey "Daten sichern" für eine interne Datensicherung des gesamten Speichers.
- Softkey "IBN-Archive"
  - Inbetriebnahmearchiv erstellen
  - Inbetriebnahmearchiv einlesen

Datenbereiche

In einem Inbetriebnahmearchiv werden folgende Datenbereiche gesichert:

Komponenten	Daten
NC-Daten	• Compile-Zyklen • Standard- und Anwender-Zyklen • Definitionen und Makros • Maschinendaten • Setting-Daten • Optionsdaten • Globale (GUD) und lokale (LUD) Anwenderdaten • Werkzeug- und Magazindaten • Schutzbereichsdaten • R-Parameter • Nullpunktverschiebungen • Kompensationsdaten • Teileprogramme • Unterprogramme • Werkstücke

Komponenten	Daten
PLC-Daten	• Hauptprogramm (MAIN) • Datenbausteine (DB) • Anwenderprogramm
Antriebsdaten	Binär-Format oder ASCII-Format
HMI-Daten	• Texte: PLC-Alarmtexte, Zyklen-Alarmtexte, Teileprogramm-Meldetexte vom Maschinenhersteller • Vorlagen: Einzelvorlagen, Werkstückvorlagen • Applikationen: Software-Applikationen, z. B. vom Maschinenhersteller • Projektierungen • Konfiguration: Konfigurationen inkl. Anzeige-Maschinendaten • Hilfe: Online-Hilfe Dateien • Versionsdaten • Protokolle: z. B. Fahrtenschreiber, Bildschirmabzüge • Programmlisten • Wörterbücher: für Chinese simplified und Chinese traditional (IME) • Datensicherungen: Kanaldaten, Achsdaten, usw. im ASCII-Format • Programme auf lokalem Laufwerk: Programme, die im Anwenderspeicherbereich der CompactFlash Card liegen.

Hinweis

Die Inbetriebnahmearchive werden mit Berücksichtigung der Datenklassen (Dateityp ARD) gespeichert. Die Antriebsdaten werden als Binärdaten gesichert, die nicht editiert werden können.

Speicherbereiche für Archive

Für die Archive stehen folgende Speicherbereiche zur Verfügung:

- auf der Anwender CompactFlash Card: beliebiges Verzeichnis.
- auf der CompactFlash Card unter:

  `/user/sinumerik/data/archive`

  oder:

  `/oem/sinumerik/data/archive`
- auf einem USB-FlashDrive

ACHTUNG
USB-FlashDrives
USB-FlashDrives sind nicht als dauerhafte Speichermedien geeignet.

11.3 So erzeugen Sie ein Inbetriebnahmearchiv

Übersicht

Die Steuerungskomponenten können wahlweise einzeln oder gemeinsam gesichert werden. Es wird empfohlen, für jede Komponente jeweils ein einzelnes Inbetriebnahmearchiv zu erstellen, um die Dateien auch unabhängig voneinander wieder einzulesen.

Voraussetzung

Es wird die Zugriffsstufe "Service" vorausgesetzt.

Inbetriebnahmearchiv erstellen

Vorgehensweise:

1. Wählen Sie den Bedienbereich "Inbetriebnahme" an.
2. Drücken Sie die Menüfortschalt-Taste und dann den Softkey "IBN-Archive".

   Das Fenster "Inbetriebnahme" wird geöffnet.
3. Wählen Sie die Option "Inbetriebnahmearchiv erstellen" und bestätigen Sie mit "OK".

   Das Fenster "Inbetriebnahmearchiv erstellen" wird geöffnet:

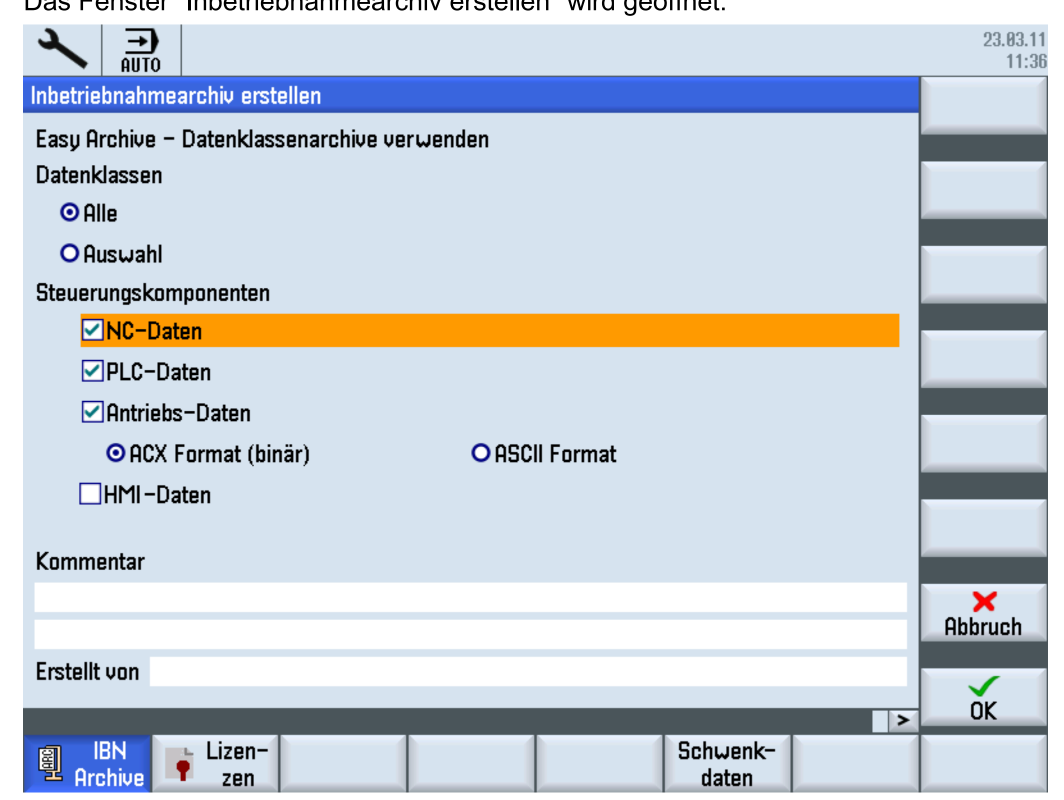


Bild 11-1 Inbetriebnahmearchiv erstellen

4. Wählen Sie mit der Taste <SELECT>, ob die Datenklassen "ignoriert" oder "berücksichtigt" werden sollen:
   - Wählen Sie "ignoriert", um alle zur Steuerungskomponente gehörigen Daten zu archivieren.
   - Wählen Sie "berücksichtigt", um nur die Daten der unter "Auswahl" angewählten Datenklasse ins Archiv zu schreiben.
5. Markieren Sie die Steuerungskomponente(n) für das Archiv.
6. Nutzen Sie die Möglichkeit, einen Kommentar und den Ersteller des Archivs einzutragen.
7. Drücken Sie den Softkey "OK", um das Archiv zu erzeugen.

   Das Fenster "Archiv erzeugen: Ablage auswählen" wird geöffnet.
8. Wählen Sie ein Verzeichnis oder drücken Sie den Softkey "Neues Verzeichnis", um ein neues Unterverzeichnis zu erstellen.

   Das Fenster "Neues Verzeichnis" wird geöffnet.
9. Geben Sie einen Namen ein und bestätigen Sie mit "OK".

   Das Verzeichnis wird unterhalb des angewählten Ordners angelegt.

   Das Fenster "Archiv erzeugen: Name" wird geöffnet.
10. Geben Sie einen Namen ein und bestätigen Sie mit "OK".

Es wird eine Archivdatei im angewählten Verzeichnis erzeugt.

11.4 So lesen Sie ein Inbetriebnahmearchiv ein

Voraussetzung

Zum Einlesen eines Archivs ist die Zugriffsstufe "Anwender" erforderlich.

Inbetriebnahmearchiv einlesen

Vorgehensweise:

1. Wählen Sie den Bedienbereich "Inbetriebnahme" an.
2. Drücken Sie die Menüfortschalt-Taste und dann den Softkey "IBN-Archive".

   Das Fenster "Inbetriebnahme" wird geöffnet.
3. Wählen Sie die Option "Inbetriebnahmearchiv einlesen" und bestätigen Sie mit "OK".

   Das Fenster "Inbetriebnahmearchiv auswählen" wird geöffnet.
4. Wählen Sie das Archiv aus und bestätigen Sie mit "OK".
5. Um das Archiv einzulesen, bestätigen Sie die Sicherheitsabfrage mit "OK".

   Das Fenster "Archiv einlesen" wird geöffnet und zeigt den Einlesevorgang mit einer Fortschrittsanzeige an.
6. Um den Einlesevorgang abzubrechen, drücken Sie den Softkey "Abbruch".

11.5 So sichern Sie nur die geänderten Maschinendaten

Voraussetzung

Es wird die Zugriffsstufe "Service" vorausgesetzt.

Differenz-Sicherung nur von geänderten Maschinendaten

Über das allgemeine MD11210 $MN_UPLOAD_MD_CHANGES_ONLY kann eingestellt werden, ob alle Maschinendaten oder nur die von der Voreinstellung abweichenden Maschinendaten im Archiv gesichert werden:

Siehe auch: MD11210 $MN_UPLOAD_MD_CHANGES_ONLY

MD11210 $MN_UPLOAD_MD_CHANGES_ONLY	
Sicherung nur von geänderten Maschinendaten	
= FFH (Voreinstellung)	
Bit 0	
. . .	
Bit 7	

Hinweis

Wirkung von MD11210

Das MD11210 $MN_UPLOAD_MD_CHANGES_ONLY ist nur bei Archiven vom Typ *.arc wirksam.

Archiv erzeugen

Vorgehensweise:

1. Stecken Sie ein Speichermedium (CompactFlash Card oder USB FlashDrive) in den entsprechenden Steckplatz an der Front der Steuerung.
2. Um ein Archiv vom Typ *.arc zu erzeugen, drücken Sie die Tasten <Ctrl> + <Alt> + C.

Ergebnis:

Es wird ein Archiv auf dem Speichermedium erzeugt:

Der Name des Archivs hat folgende Struktur: CompleteArchive_Date_Time.arc

Beispiel:

CompleteArchive2010-08-11_08-36-15.arc steht für ein am 11.08.2010 um 8:36:15 Uhr erzeugtes Archiv.

11.6 Beispiel: Datenarchivierung "Easy Archive" (Use case)

Easy Archive

Die SINUMERIK 828D besitzt mit "Easy Archive" ein grundlegend neues Verfahren der Datenarchivierung. Dieses Verfahren ist exakt auf die Belange bei der Herstellung von Serienmaschinen zugeschnitten. "Easy Archive" basiert auf einer strikten Trennung zwischen SINUMERIK Systemsoftware, Anpassdaten des OEMs (Maschinendaten, Herstellerzyklen) und den Daten des Bedieners (Teileprogramme, Werkzeugkorrekturen). Die Anpassdaten unterliegen einer weiteren Trennung zwischen solchen Daten, die bei allen Maschinen eines Typs gleich sind, und denen, die eine maschinenindividuelle Anpassung erfahren.

Dies wird anhand eines Beispiels verdeutlicht:

Anwenderbeispiel

Ein Maschinenhersteller baut ein vertikales Bearbeitungszentrum in Serie. Die Anpassdaten werden an einer sogenannten Prototypmaschine erstellt. Der Anpassdatensatz dieser Prototypmaschine wird später auf alle Serienmaschinen portiert (geklont). Nach dem Portieren der Daten werden jedoch an jeder Maschine individuelle Einstellungen vorgenommen.

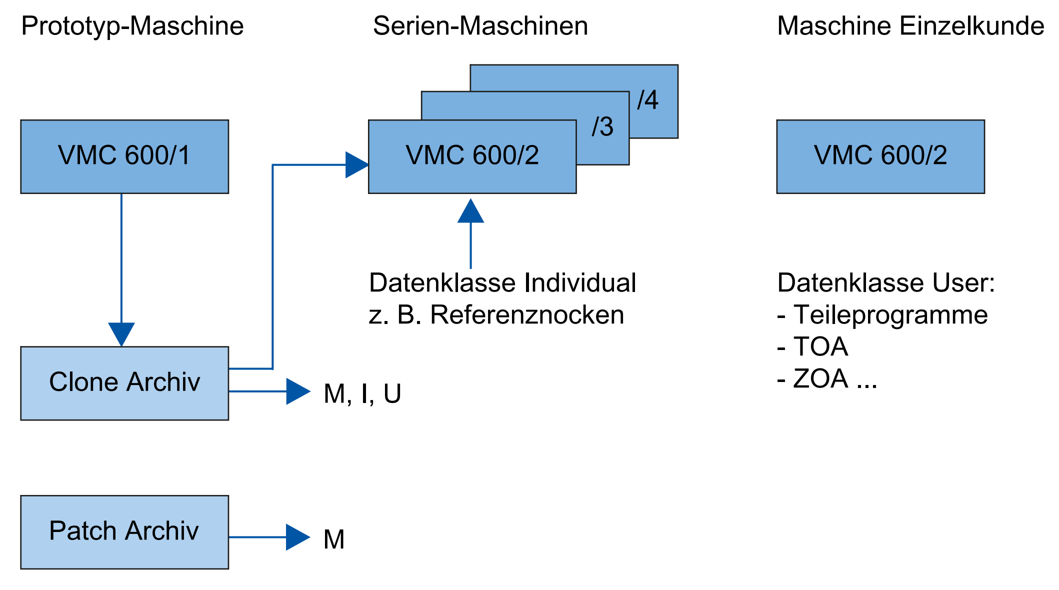


Bild 11-2 Anwenderbeispiel

Beispielsweise werden die Referenznocken sowie die Kugelrollspindeln vermessen und als individuelle Anpassdaten eingetragen. Tritt nun ein Fehler beim Endanwender auf, so wird der Fehler an der Prototypmaschine reproduziert und behoben. Wird nun ein komplettes Archiv der Prototypmaschine an die betroffene Maschine übertragen, so werden die individuellen Anpassdaten dieser Maschine mit den individuellen Anpassdaten der Prototypmaschine überschrieben.

Bei der SINUMERIK 828D können die Anpassdaten des Maschinenherstellers, die keiner individuellen Änderung unterliegen separat archiviert werden. Wird dieses Archiv auf die betroffene Maschine übertragen, so ist sicher gestellt, dass die individuellen Anpassdaten sowie die Daten des Endanwenders erhalten bleiben. Somit wird der Update-Vorgang des Maschinenherstellers wesentlich vereinfacht.

Vorteile

Der Vorteil von "Easy Archive" liegt darin, dass die Archiverstellung direkt an der Bedienoberfläche der SINUMERIK 828D erfolgt. Es wird also kein separater PC für die Archive benötigt.

Durch die Trennung der Systemdaten von den Anpass- und Anwenderdaten werden die System-Updates der SINUMERIK 828D komplett ohne Änderungen der Anpassdaten seitens des OEMs durchgeführt. Ein System-Update kann vom Endanwender in kurzer Zeit selbst ausgeführt werden.

11.7 Serielle Schnittstelle parametrieren

Datenaustausch

Der Datenaustausch über die serielle Schnittstelle ist von folgenden Bedienbereichen aus möglich:

- Bedienbereich "Programm-Manager"
- Bedienbereich "Inbetriebnahme" → Softkey "Systemdaten"

Um die Schnittstellenparameter einzustellen, betätigen Sie folgende Softkeys:



Hinweis

Falls die Schnittstelle bereits belegt ist, z. B. weil ein Modem angeschlossen ist, ist kein Datenaustausch über die serielle Schnittstelle möglich und es wird eine Meldung ausgegeben.

Parameterbeschreibung

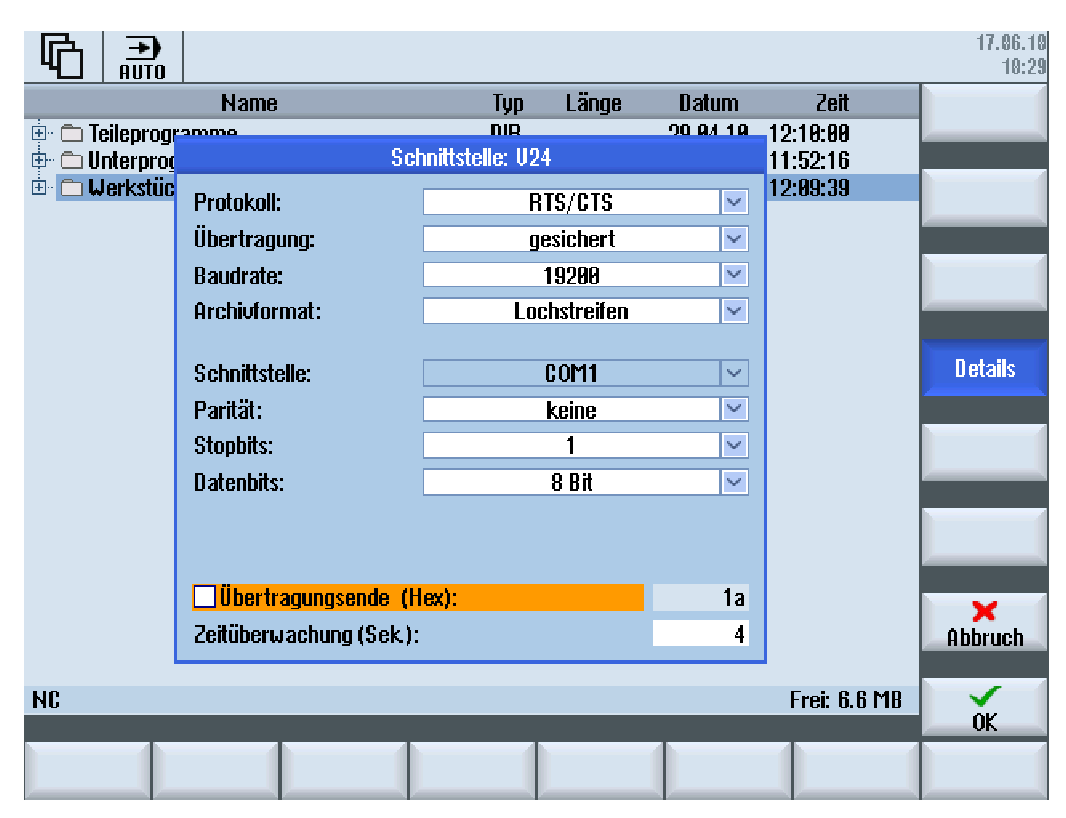


Bild 11-3 Parameter einstellen

Parameter	Zulässige Werte	
Protokoll:	**RTS/CTS (Voreinstellung)** Xon/Xoff	
Übertragung:	normal **gesichert (Voreinstellung)**	
Baudrate:	**19200 (Voreinstellung)** 300 ... max. 19200	
Archivformat:	Lochstreifen Binär-Format (PC-Format) → nur mit RTS/CTS Protokoll	
Schnittstelle:	COM1	
Parität:	**keine (Voreinstellung)** gerade ungerade	
Stopbits:	**1 (Voreinstellung)** 2	
Datenbits:	5 Bit 6 Bit 7 Bit **8 Bit (Voreinstellung)**	
Xon (nur bei Einstellung Xon/Xoff)		11
Xoff (nur bei Einstellung Xon/Xoff)		13
Übertragungsende (Hex)		1a
Zeitüberwachung (Sek.)		4

V24 deaktivieren

Um die Schnittstelle für den Datenaustausch zu deaktivieren, ist folgendes Maschinendatum zu setzen:

- MD51233 $MNS_ENABLE_GSM_MODEM = 0 (Voreinstellung)

  Die V24-Schnittstelle ist für den Datentransfer freigegeben.
- MD51233 $MNS_ENABLE_GSM_MODEM = 1

  Danach werden die Softkeys für den Datentransfer nicht angezeigt.

  Mit dieser Einstellung ist das GSM-Modem aktiviert.

Anhang A

A.1 Hinweis zur verwendeten Fremdsoftware

This software product (LICENSED PRODUCT), implementing the Object Management Group's "Internet Inter-ORB Protocol", is protected by copyright and is distributed under the following license restricting its use. Portions of LICENSED PRODUCT may be protected by one or more U.S. or foreign patents, or pending applications.

LICENSED PRODUCT is made available for your use provided that you include this license and copyright notice on all media and documentation and the software program in which this product is incorporated in whole or part. You may copy, modify, distribute, or sublicense the LICENCED PRODUCT without charge as part of a product or software program developed by you, so long as you preserve the functionality of interoperating with the Object Management Group's "Internet Inter-ORB Protocol" version one. However, any uses other than the foregoing uses shall require the express written consent of Sun Microsystems, Inc.

The names of Sun Microsystems, Inc. and any of its subsidiaries or affiliates may not be used in advertising or publicity pertaining to distribution of the LICENSED PRODUCT as permitted herein. This license is effective until terminated by Sun for failure to comply with this license. Upon termination, you shall destroy or return all code and documentation for the LICENSED PRODUCT.

LICENSED PRODUCT IS PROVIDED AS IS WITH NO WARRANTIES OF ANY KIND INCLUDING THE WARRANTIES OF DESIGN, MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR ARISING FROM A COURSE OF DEALING, USAGE OR TRADE PRACTICE.

LICENSED PRODUCT IS PROVIDED WITH NO SUPPORT AND WITHOUT ANY OBLIGATION ON THE PART OF SUN OR ANY OF ITS SUBSIDIARIES OR AFFILIATES TO ASSIST IN ITS USE, CORRECTION, MODIFICATION OR ENHANCEMENT. SUN OR ANY OF ITS SUBSIDIARIES OR AFFILIATES SHALL HAVE NO LIABILITY WITH RESPECT TO THE INFRINGEMENT OF COPYRIGHTS, TRADE SECRETS OR ANY PATENTS BY LICENSED PRODUCT OR ANY PART THEREOF.

IN NO EVENT WILL SUN OR ANY OF ITS SUBSIDIARIES OR AFFILIATES BE LIABLE FOR ANY LOST REVENUE OR PROFITS OR OTHER SPECIAL, INDIRECT AND CONSEQUENTIAL DAMAGES, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Hinweis

Weitere Hinweise zur verwendeten Fremdsoftware finden Sie auf der Produkt DVD in der Datei Readme_OSS.

A.2 Liste der Sprachkennzeichen für Dateinamen

Unterstützte Sprachen

Standard-Sprachen:

Sprache	Abkürzung im Dateinamen
Chinesisch simplified	chs
Chinesisch traditional	cht
Deutsch	deu
Englisch	eng
Spanisch	esp
Französisch	Fra
Italienisch	Ita
Koreanisch	kor
Portugiesisch (Brasilien)	ptb

Weitere Sprachen:

Sprache	Abkürzung im Dateinamen
Tschechisch	csy
Dänisch	dan
Finnisch	fin
Ungarisch	hun
Japanisch	jpn
Niederländisch	nld
Polnisch	plk
Rumänisch	rom
Russisch	rus
Slowakisch	sky
Slowenisch	slv
Schwedisch	sve
Türkisch	trk

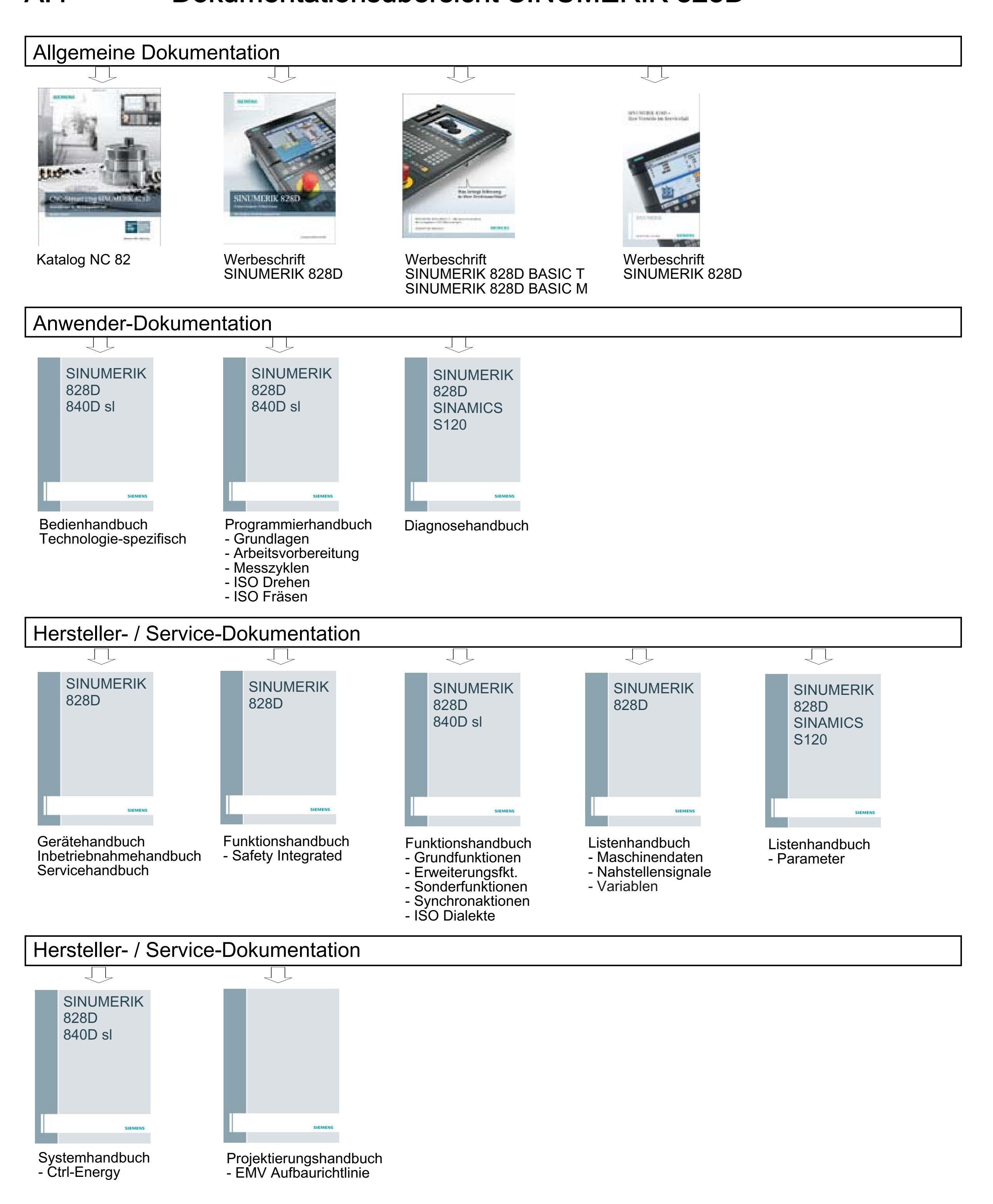
A.3 Liste der Abkürzungen

Abkürzung	Bedeutung	Erklärung
ALM	Active Line Module	
ASCII	American Standard Code for Information Interchange	Amerikanische Code Norm für den Informationsaustausch
AUTO	Betriebsart "Automatic"	
BAG	Betriebsartengruppe	
BTSS	Bedientafelschnittstelle	
BERO	Berührungsloser Endschalter mit Rückgekoppelter Oszillation	
BICO	Binector Connector	Verschaltungstechnik beim Antrieb
CEC	Cross Error Compensation	
CNC	Computerized Numerical Control	Computerunterstützte Numerische Steuerung
dbSI	drive based Safety Integrated	im Antrieb integrierte Sicherheitsfunktionen
DB	Datenbaustein in der PLC	
DBB	Datenbausteinbyte in der PLC	
DBW	Datenbausteinwort in der PLC	
DBX	Datenbausteinbit in der PLC	
DDE	Dynamic Data Exchange	Dynamischer Datenaustausch
DDS	Drive Data Set	Antriebsdatensatz
DIN	Deutsche Industrie Norm	
DO	Drive object	Antriebsobjekt
DRAM	Dynamic Random Access Memory	Dynamischer Speicherbaustein
DRF	Differential Resolver Function	Differential-Drehmelder-Funktion (Handrad)
DRY	Dry Run	Probelauf-Vorschub
EDS	Encoder Data Set	Geberdatensatz
ESR	Erweitertes Stillsetzen und Rückziehen	
FIFO	First In - First Out	Verfahren, wie Daten in einem Speicher abgelegt und wieder abgerufen werden
GUD	Global User Data	Globale Anwenderdaten
HD	Hard Disk	Festplatte
HW	Hardware	
HSC	High Speed Cutting	
IME	Input Method Editor	Eingabe asiatischer Zeichen
INC	Increment	Schrittmaß
INI	Initializing Data	Initialisierungsdaten
IGBT	Insulated Gate Bipolar Transistor	
IPO	Interpolator	
IRT	Isochronous Real Time	Taktsynchrone Kommunikation
ISO	International Standardization Organisation	Internationale Organisation für Normung
JOG	Betriebsart "Jogging":	Tippen über die Richtungstasten
LEC	Leadscrew Error Compensation	Spindelsteigungsfehler-Kompensation

Abkürzung	Bedeutung	Erklärung
LED	Light Emitting Diode	Leuchtdiode
LUD	Local User Data	Lokale Anwenderdaten
MB	Megabyte	
MCP	Machine Control Panel	Maschinensteuertafel
MD	Maschinendaten	
MDA	Betriebsart "Manual Data Automatic"	Handeingabe
MDS	Motor Data Set	Motordatensatz
MKS	Maschinen-Koordinatensystem	
MLFB	Maschinenlesbare Fabrikatbezeichnung	
MPF	Main Program File	Hauptprogramm (NC-Teileprogramm)
MAIN	Main program	Hauptprogramm (OB1, PLC)
MPI	Multi Point Interface	Mehrpunktfähige Schnittstelle
NCK	Numerical Control Kernel	Zentraleinheit der Numerischen Steuerung
NCU	Numerical Control Unit	Hardware Einheit des NCK
NPV	Nullpunktverschiebung	
OEM	Original Equipment Manufacturer	
PCU	Programmable Control Unit	
PI	Programm Instanz	
PG	Programmiergerät	
PLC	Programmable Logic Control	Speicherprogrammierbare Steuerung
POE	Programmorganisationseinheit	im PLC-Anwenderprogramm
PPU	Panel Processing Unit	Steuerung auf Panel-Basis
PZD	Prozessdaten für Antriebe	
QEC	Quadrant Error Compensation	Quadrantenfehler-Kompensation
REF POINT	"Referenzpunkt fahren" in der Betriebsart JOG	
REPOS	"Repositionieren" in der Betriebsart JOG	
RPA	R-Parameter Active	Speicherbereich in NCK für R-Parameternummern
RTC	Real Time Clock	Echtzeituhr
SBL	Single Block	Einzelsatz
SBR	Subroutine	Unterprogramm (PLC)
SD	Setting-Datum	
SDB	System-Datenbaustein	
SEA	Setting Data Active	Kennzeichnung (Dateityp) für Setting-Daten
SK	Softkey	
SLM	Smart Line Module	
SPF	Subprogram file	Unterprogramm (NC)
SRAM	Static Random Access Memory	Statischer Speicherbaustein
SW	Software	
TEA	Testing Data Aktive	Kennung für Maschinendaten
TO	Tool Offset	Werkzeugkorrektur

Abkürzung	Bedeutung	Erklärung
TOA	Tool Offset Active	Kennzeichnung (Dateityp) für Werkzeugkorrekturen
VPM	Voltage Protection Module	
VSM	Voltage Sensing Module	
WKS	Werkstück-Koordinatensystem	
WZMG	Werkzeugmagazinverwaltung	
WZV	Werkzeugverwaltung	
ZOA	Zero Offset Active	Kennzeichnung (Dateityp) für Nullpunktverschiebungsdaten

A.4 Dokumentationsübersicht SINUMERIK 828D



Glossar

Active Line Module (ALM)

Geregelte, selbstgeführte Ein-/Rückspeiseeinheit (mit IGBTs in Ein- und Rückspeiserichtung), die die DC-Zwischenkreisspannung für die → Motor Module zur Verfügung stellt.

Antrieb

Ein Antrieb ist die Gesamtheit von Motor (elektrisch oder hydraulisch), Stellglied (Umrichter, Ventil), Regelung, Mess-System und Versorgung (Einspeisung, Druckspeicher). Bei elektrischen Antrieben wird zwischen Umrichter- oder Wechselrichtersystem unterschieden. Beim Umrichtersystem (z. B. → MICROMASTER 4) sind aus Anwendersicht Einspeisung, Stellglied und Regelung in einem Gerät zusammengefasst; beim Wechselrichtersystem (z. B. → SINAMICS S) wird die Versorgung mittels → Line Module ausgeführt, damit ein Zwischenkreis realisiert, an dem die → Wechselrichter (→ Motor Module) angeschlossen werden. Die Regelung (→ Control Unit) ist in einem separaten Gerät untergebracht und über → DRIVE CLiQ mit den übrigen Komponenten verbunden.

Antriebsgerät

Gesamtheit aller über → DRIVE CLiQ verbundenen Komponenten, die zur Realisierung einer Antriebsaufgabe notwendig sind: → Motor Module → Control Unit → Line Module sowie die erforderliche → Firmware und die → Motoren, jedoch ohne ergänzende Komponenten wie Filter und Drosseln. In einem Antriebsgerät können mehrere → Antriebe realisiert sein. Siehe → Antriebssystem

Antriebskomponente

Hardware-Komponente, die an eine → Control Unit über → DRIVE CLiQ oder anders angeschlossen ist. Antriebskomponenten sind z. B.: → Motor Modules, → Line Modules, → Motoren, → Sensor Modules und → Terminal Modules. Die Gesamtanordnung einer Control Unit mitsamt den angeschlossenen Antriebskomponenten heißt → Antriebsgerät.

Antriebsobjekt (Drive Object DO)

Ein Antriebsobjekt ist eine eigenständige in sich geschlossene Softwarefunktionalität, die ihre eigenen → Parameter und evtl. auch ihre eigenen → Störungen und → Warnungen hat. Die Antriebsobjekte können standardmäßig vorhanden sein (z. B. On Board I/O), einfach anlegbar (z. B. Terminal Board 30, TB30) oder auch mehrfach anlegbar sein (z. B. → Servoregelung). Jedes Antriebsobjekt hat in der Regel sein eigenes Fenster für seine Parametrierung und Diagnose.

Antriebsparameter

Parameter einer Antriebsachse, die z. B. die Parameter der zugehörigen Regler und die Motor- und Geberdaten enthalten. Die Parameter der übergeordneten Technologiefunktionen (Positionieren, Hochlaufgeber) werden im Gegensatz hierzu als Applikationsparameter bezeichnet.

Antriebssystem

Ein Antriebssystem ist die Gesamtheit der zu einem Antrieb gehörenden Komponenten einer Produktfamilie, z. B. SINAMICS. Ein Antriebssystem beinhaltet z. B. → Line Module, - > Motor Module, → Geber, → Motoren, → Terminal Module und → Sensor Module sowie ergänzende Komponenten wie Drosseln, Filter, Leitungen usw.

Antriebsverband

Ein Antriebsverband besteht aus einer → Control Unit und den daran über → DRIVE CLiQ angeschlossenen → Motor Module und → Line Module.

Anwendersichten

Anwendersichten sind anwenderspezifische Sammlungen von Maschinendaten. Sie dienen dazu, alle in einem bestimmten Bedienzustand relevanten Maschinendaten aus verschiedenen Bereichen zur Bearbeitung zur Verfügung zu stellen.

Die Anwendersichten werden unter folgendem Pfad auf der CompactFlash Card gespeichert:

```
user/sinumerik/hmi/template/user_views
```

Als Vorlage sind bereits folgende Anwendersichten vorhanden:

- Electrical_Startup
- Mechanical_Startup
- Optimizing_Axis

Basic Infeed

Gesamtfunktionalität einer Einspeisung mit → Basic Line Module inklusive der benötigten Zusatzkomponenten (Filter, Schaltgeräte, usw.).

Basic Line Module

Ungeregelte Einspeiseeinheit (Diodenbrücke oder Thyristorbrücke ohne Rückspeisung) zur Gleichrichtung der Netzspannung für den → Zwischenkreis.

Certificate of License (CoL)

Das CoL ist der Nachweis der → Lizenz. Das Produkt darf nur durch den Inhaber der → Lizenz oder beauftragten Personen genutzt werden. Auf dem CoL befinden sich unter anderem folgende für das Lizenzmanagement relevante Daten:

- Produktname
- → Lizenznummer
- Lieferscheinnummer
- → Hardware-Seriennummer

CompactFlash Card

Die CompactFlash Card repräsentiert als Träger aller remanenten Daten einer SINUMERIK solution line Steuerung die Identität dieser Steuerung. Die CompactFlash Card ist eine Speicherkarte, die in die → Control Unit von außen steckbar ist. Auf der CompactFlash Card befinden sich folgende für das Lizenzmanagement relevante Daten:

- → Hardware-Seriennummer
- Lizenzinformationen einschließlich → License Key

Control Unit

Zentrale Steuerungs- und Regelungsbaugruppe. Es gibt folgende Control Units:

- SIMOTION Control Units, z. B. D425 und D435
- SINAMICS Control Units, z. B. CU320
- SINUMERIK solution line Control Units, z. B. NCU, PPU

Datenklasse Individual (I)

Diese Datenklasse umfasst die Daten, die eine bestimmte Maschine betreffen und im Rahmen der Inbetriebnahme ebenfalls vom OEM oder zu einem späteren Zeitpunkt beim Händler erzeugt werden. Diese Datenklasse wird im weiteren Dokument mit "I" bezeichnet.

Datenklasse Manufacturer (M)

Diese Datenklasse umfasst alle Daten, die vom Maschinenhersteller (OEM) bei der Erstinbetriebnahme einer Maschine einer Baureihe festgelegt werden, und wird im weiteren Dokument mit "M" bezeichnet.

Datenklasse System (S)

Diese Datenklasse umfasst die Daten in den Verzeichnissen Siemens und System auf der CompactFlash Card und wird im weiteren Dokument mit "S" bezeichnet.

Datenklasse User (U)

Diese Datenklasse umfasst alle Daten des Anwenders sowie die Daten, die zur Laufzeit der Maschine entstehen, beispielsweise der Wartungsintervall-Timer. Diese Datenklasse wird im weiteren Dokument mit "U" bezeichnet.

Double Motor Module (DMM)

An ein Double Motor Module können zwei Motoren angeschlossen und betrieben werden. Siehe → Motor Module.

DRIVE CLiQ

Abkürzung für "Drive Component Link with IQ".

Kommunikationssystem zum Verbinden der verschiedenen Komponenten eines SINAMICS Antriebssystems, wie z. B. → Control Unit, → Line Module, → Motor Module, → Motoren und Drehzahl-/Lagegeber.

DRIVE CLiQ basiert auf Industrial Ethernet mit Twisted-Pair-Leitungen. Zusätzlich zu den Sende- und Empfangssignalen wird auch die +24 V-Spannungsversorgung über die DRIVE CLiQ-Leitung zur Verfügung gestellt.

Einspeisung

Eingangsteil einer Umrichteranlage zur Erzeugung einer DC-Zwischenkreisspannung zur Speisung eines oder mehrerer → Motor Modules inklusive aller dafür benötigten Komponenten wie → Line Modules, Sicherungen, Drosseln, Netzfilter und Firmware sowie - falls erforderlich - anteiliger Rechenleistung in einer → Control Unit.

Externer Geber

Lagegeber, der nicht in oder an den → Motor eingebaut, sondern außen an die Arbeitsmaschine oder über ein mechanisches Zwischenglied angebaut ist. Der externe Geber (Anbaugeber) wird zur direkten Lageerfassung verwendet.

Geber

Lagepositionen erfasst und zur elektronischen Verarbeitung bereitstellt. Je nach mechanischer Ausführung können Geber in den → Motor eingebaut (→ Motorgeber) oder an die externe Mechanik angebaut (→ externer Geber) werden. Nach der Bewegungsart wird unterschieden zwischen rotatorischen Gebern (manchmal auch Drehgeber genannt) und translatorischen Gebern (z. B. → Linearmaßstab). Nach der Messwertbereitstellung wird unterschieden zwischen → Absolutwertgebern (Codegeber) und → Inkrementalgebern. Siehe → Inkrementalgeber TTL/HTL → Inkrementalgeber sin/cos 1 Vpp → Resolver

Hardware

Als Hardware im Rahmen des Lizenzmanagements von SINUMERIK → Softwareprodukten wird die Komponente einer SINUMERIK Steuerung bezeichnet, der aufgrund ihrer eindeutigen Kennung → Lizenzen zugeordnet werden. Auf dieser Komponente werden auch die Lizenzinformationen remanent gespeichert, z. B. auf einer → CompactFlash Card.

Hardware-Seriennummer

Die Hardware-Seriennummer ist unveränderlicher Bestandteil der → CompactFlash Card. Über sie wird eine Steuerung eindeutig identifiziert. Die Hardware-Seriennummer kann ermittelt werden über:

- → Certificate of License
- Bedienoberfläche
- Aufdruck auf der → CompactFlash Card

License Key

Der License Key ist der "technische Repräsentant" der Summe aller → Lizenzen, die einer bestimmten, durch ihre → Hardware-Seriennummer eindeutig gekennzeichneten → Hardware, zugeordnet sind.

Line Module

Ein Line Module ist ein Leistungsteil, das aus einer dreiphasigen Netzspannung die Zwischenkreisspannung für ein oder mehrere → Motor Modules erzeugt. Bei SINAMICS gibt es die folgenden drei Arten von Line Modules: → Basic Line Module, → Smart Line Module und → Active Line Module.

Die Gesamtfunktion einer Einspeisung inklusive der benötigten Zusatzkomponenten wie → Netzdrossel, anteilige Rechenleistung in einer → Control Unit, Schaltgeräten usw. heißt → Basic Infeed, → Smart Infeed und → Active Infeed.

Lizenz

Eine Lizenz wird als Recht zur Nutzung eines → Softwareproduktes vergeben. Die Repräsentanten dieses Rechtes sind:

- → Certificate of License (CoL)
- → License Key

Lizenznummer

Die Lizenznummer ist das Merkmal einer → Lizenz, über das sie eindeutig identifiziert wird.

Motor

Die von → SINAMICS ansteuerbaren Elektromotoren werden grob bezüglich der Bewegungsrichtung in rotatorisch und linear; und bezüglich des elektromagnetischen Funktionsprinzips in synchron und asynchron eingeteilt. Bei SINAMICS werden die Motoren an ein → Motor Modul angeschlossen. Siehe → Synchronmotor → Asynchronmotor → Motorgeber → Externer Geber

Motor Module

Ein Motor Module ist ein Leistungsteil (DC-AC Wechselrichter), das die Energie für die angeschlossenen Motor(en) zur Verfügung stellt. Die Energieversorgung erfolgt durch bei den → Zwischenkreis des → Antriebsgerätes. Ein Motor Modul muss über → DRIVE CLiQ mit einer → Control Unit verbunden werden, in der die Steuer- und Regelungsfunktionen für das Motor Modul hinterlegt sind. Es gibt Single Motor Module und → Double Motor Module.

Motorgeber

In den Motor integrierter oder an den Motor angebauter → Geber, z. B. Resolver, → Inkrementalgeber TTL/HTL oder → Inkrementalgeber sin/cos 1 Vpp. Der Geber dient zur Erfassung der Motordrehzahl. Bei Synchronmotoren zusätzlich auch zur Erfassung des Rotorlagewinkels (des Kommutierungswinkels für die Motorströme). Bei Antrieben ohne zusätzliches → direktes Lagemess-System wird er auch als → Lagegeber zur Lageregelung verwendet. Zusätzlich zu den Motorgebern gibt es noch die → externen Geber zur → Direkten Lageerfassung.

Option

Eine Option ist ein SINUMERIK → Softwareprodukt, dass nicht in der Grundausführung enthalten ist und für dessen Nutzung eine → Lizenz erworben werden muss.

Parameter

Veränderliche Größe innerhalb des Antriebssystems, die der Anwender lesen und teilweise auch schreiben kann. Bei → SINAMICS erfüllt ein Parameter alle Festlegungen, die für Antriebsparameter im → PROFIdrive-Profil festgelegt sind. Siehe → Beobachtungsparameter → Einstellparameter

Produkt

Ein Produkt ist im Rahmen des Lizenzmanagements von SINUMERIK → Softwareprodukten durch folgende Daten gekennzeichnet:

- Produktbezeichnung
- Bestellnummer
- → Lizenznummer

Sensor Module

Hardware-Modul zur Auswertung von Drehzahl-/Lagegeber-Signalen und Bereitstellung der ermittelten Istwerte als numerische Werte an einer → DRIVE CLiQ-Buchse. Es gibt 3 mechanische Varianten von Sensor Modules:

- SMCxx = Sensor Module Cabinet-Mounted = Sensor Module für die Aufschnappmontage im Schaltschrank.
- SME = Sensor Module Externally Mounted = Sensor Module mit hoher Schutzart zur Montage außerhalb des Schaltschranks.

Servoantrieb

Ein elektrischer Servoantrieb besteht aus einem Motor, einem → Motor Module und einer → Servoregelung sowie in den meisten Fällen aus einem Drehzahl- und Lage-Geber. Elektrische Servoantriebe arbeiten in der Regel sehr präzise und mit einer hohen Dynamik. Sie sind für Taktzeiten bis zu 100 ms geeignet. Sie haben häufig eine sehr hohe kurzzeitige Überlastbarkeit und ermöglichen dadurch extrem schnelle Beschleunigungsvorgänge. Servoantriebe gibt es als rotatorische und als Linearantriebe.

Servoregelung

Diese Regelungsart ermöglicht für → Motoren mit → Motorgebern einen Betrieb mit hoher Genauigkeit und Dynamik. Neben der Drehzahlregelung kann auch eine Lageregelung enthalten sein.

Smart Line Module (SLM)

Ungeregelte Ein-/Rückspeiseeinheit mit Diodenbrücke für die Einspeisung und kippsicherer, netzgeführte Rückspeisung über IGBTs. Das Smart Line Module stellt die DC-Zwischenkreisspannung für die → Motor Module zur Verfügung.

Softwareprodukt

Als Softwareprodukt wird allgemein ein Produkt bezeichnet, das auf einer → Hardware zur Bearbeitung von Daten installiert wird. Im Rahmen des Lizenzmanagements von SINUMERIK Softwareprodukten wird für die Nutzung jedes Softwareproduktes eine entsprechende → Lizenz benötigt.

Index

N

O

P

Q

R

S

T

U